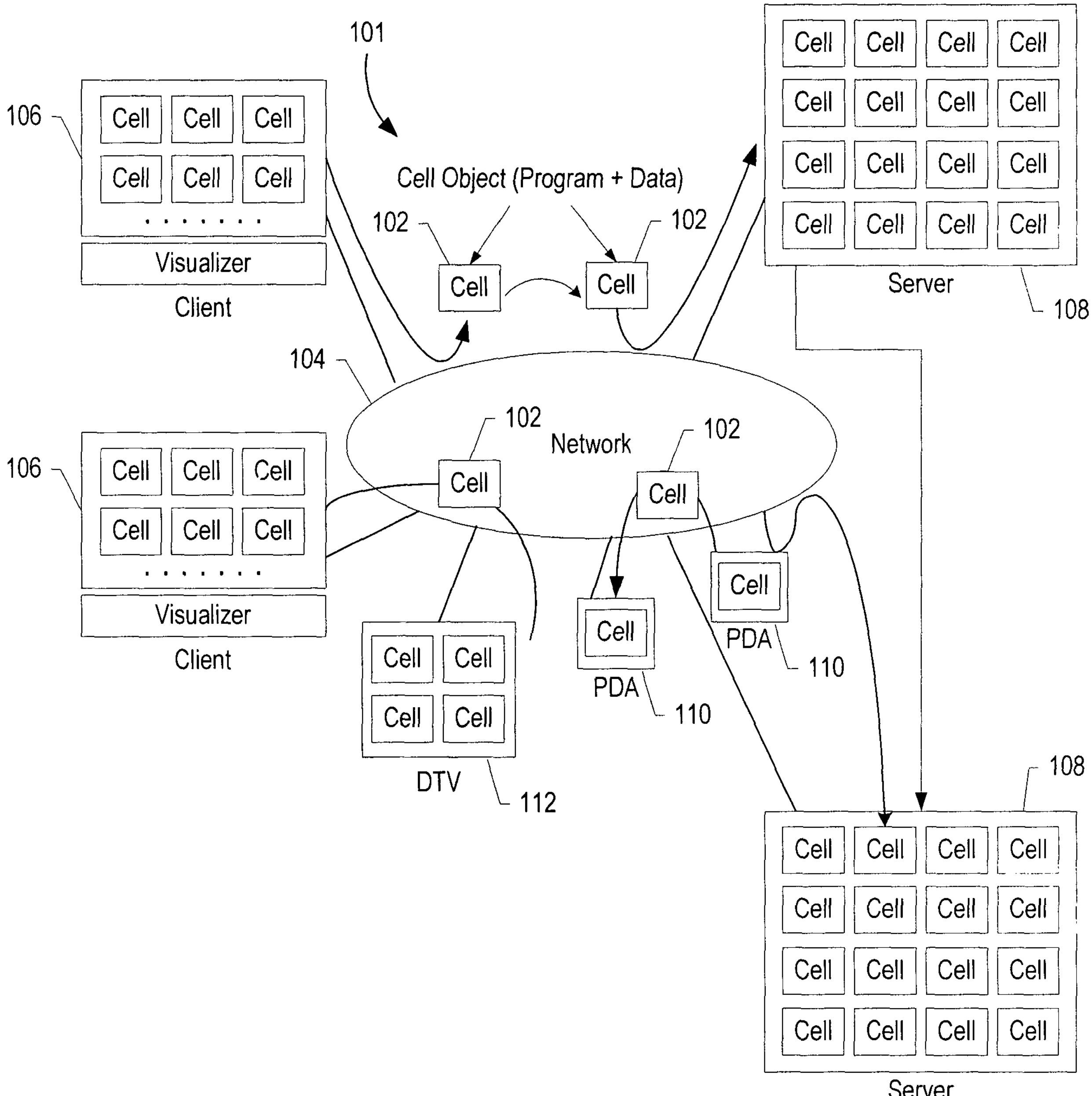
101
Cell Object (Program + Data)
106
Cell
Cell
Cell
Cell
Cell
Cell
Visualizer
Client
102
Cell
102
Cell
Cell
Cell
Cell
Cell
Cell
Cell
Cell
Cell
Cell
Cell
Cell
Cell
Cell
Cell
Cell
Cell
Server
108
104
Network
102
Cell
102
Cell
106
Cell
Cell
Cell
Cell
Cell
Cell
Visualizer
Client
Cell
Cell
Cell
Cell
DTV
112
Cell
PDA
110
Cell
PDA
110
108
Cell
Cell
Cell
Cell
Cell
Cell
Cell
Cell
Cell
Cell
Cell
Cell
Cell
Cell
Cell
Cell
Server

Figure 1

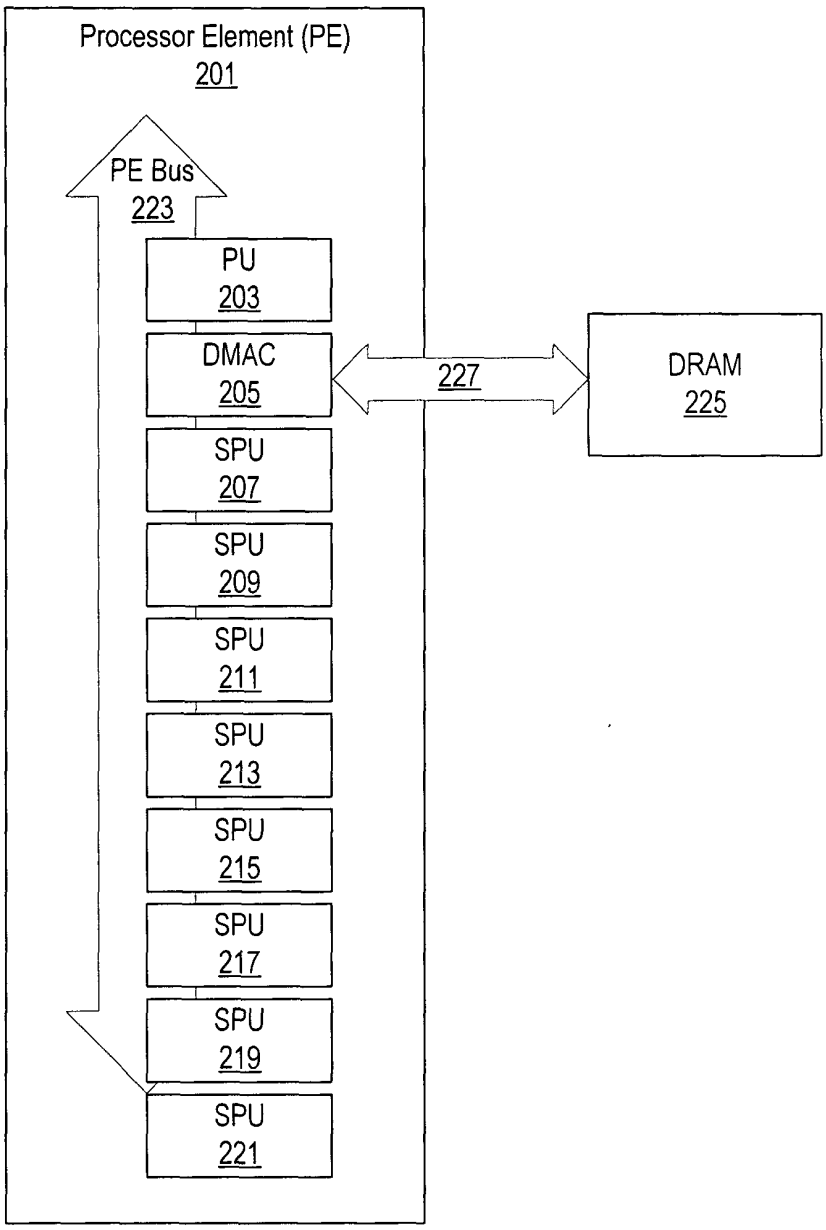
Processor Element (PE)
201
PE Bus
223
PU
203
DMAC
205
227
DRAM
225
SPU
207
SPU
209
SPU
211
SPU
213
SPU
215
SPU
217
SPU
219
SPU
221

Figure 2

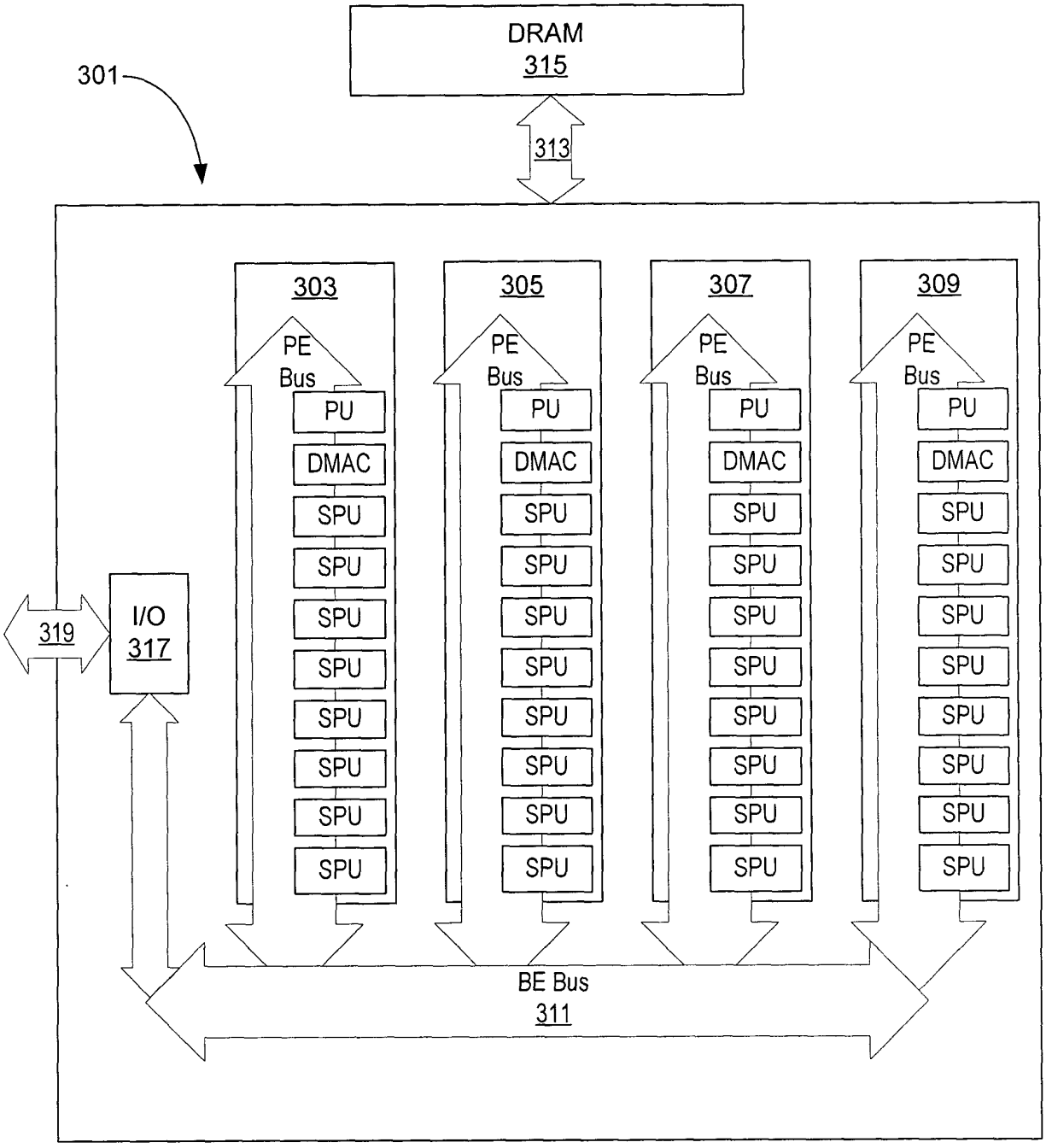
DRAM
315
301
313
303
305
307
309
PE
Bus
PU
DMAC
SPU
I/O
317
319
BE Bus
311

Figure 3

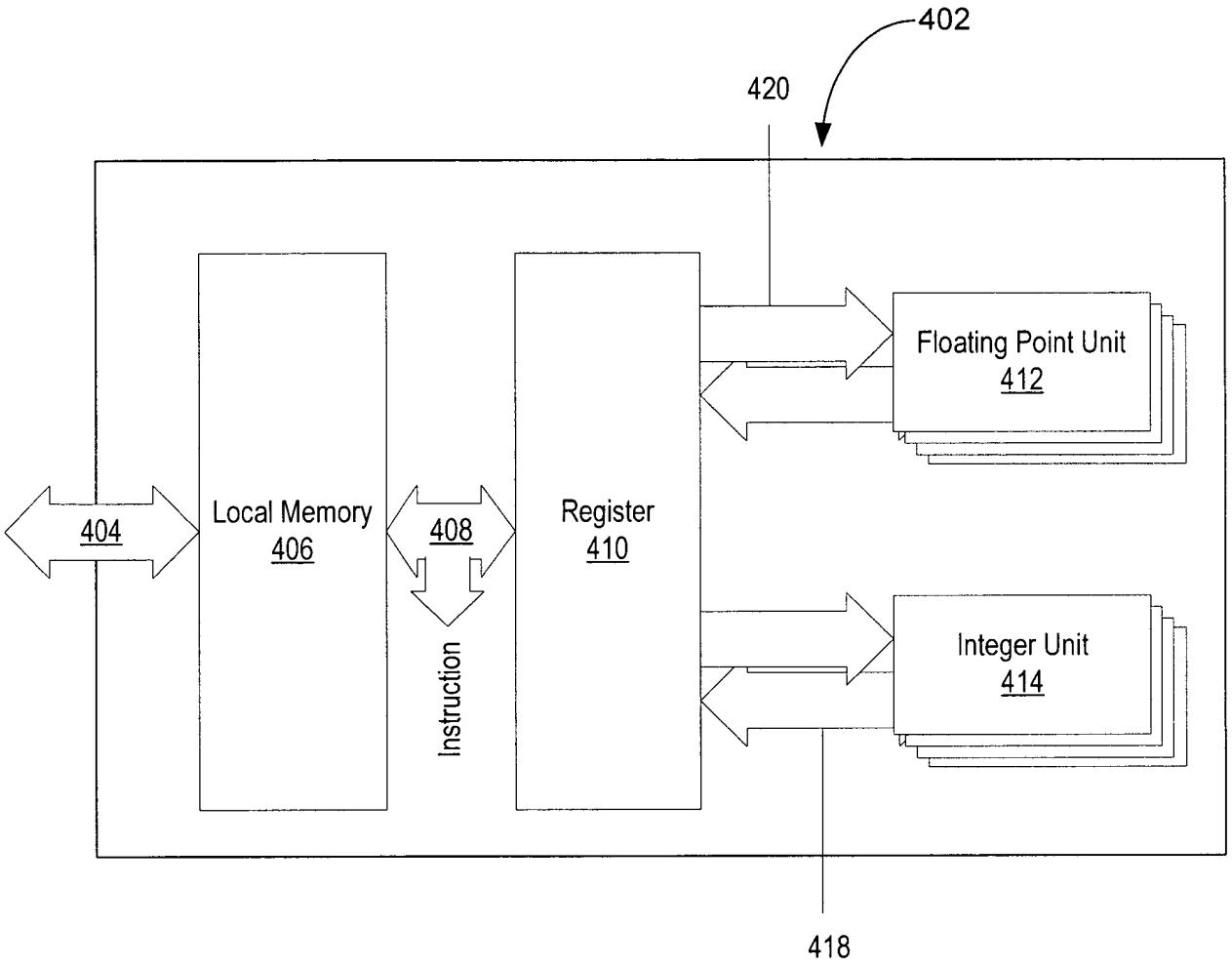
402
420
Floating Point Unit
412
404
Local Memory
406
408
Register
410
Instruction
Integer Unit
414
418

Figure 4

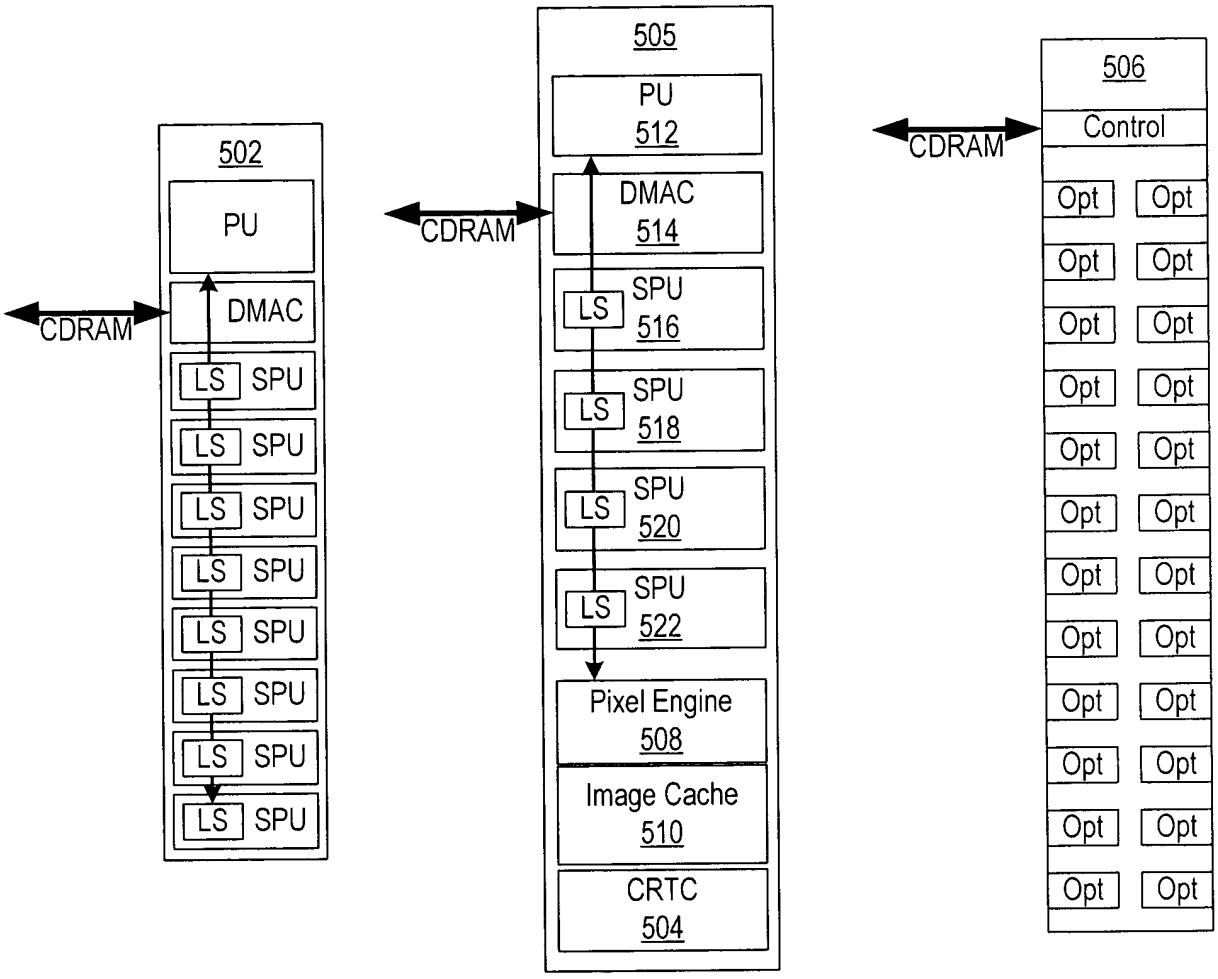
502
PU
DMAC
CDRAM
LS SPU
LS SPU
LS SPU
LS SPU
LS SPU
LS SPU
LS SPU
LS SPU
505
PU
512
DMAC
514
CDRAM
LS SPU
516
LS SPU
518
LS SPU
520
LS SPU
522
Pixel Engine
508
Image Cache
510
CRTC
504
506
Control
CDRAM
Opt Opt
Opt Opt
Opt Opt
Opt Opt
Opt Opt
Opt Opt
Opt Opt
Opt Opt
Opt Opt
Opt Opt
Opt Opt

Figure 5

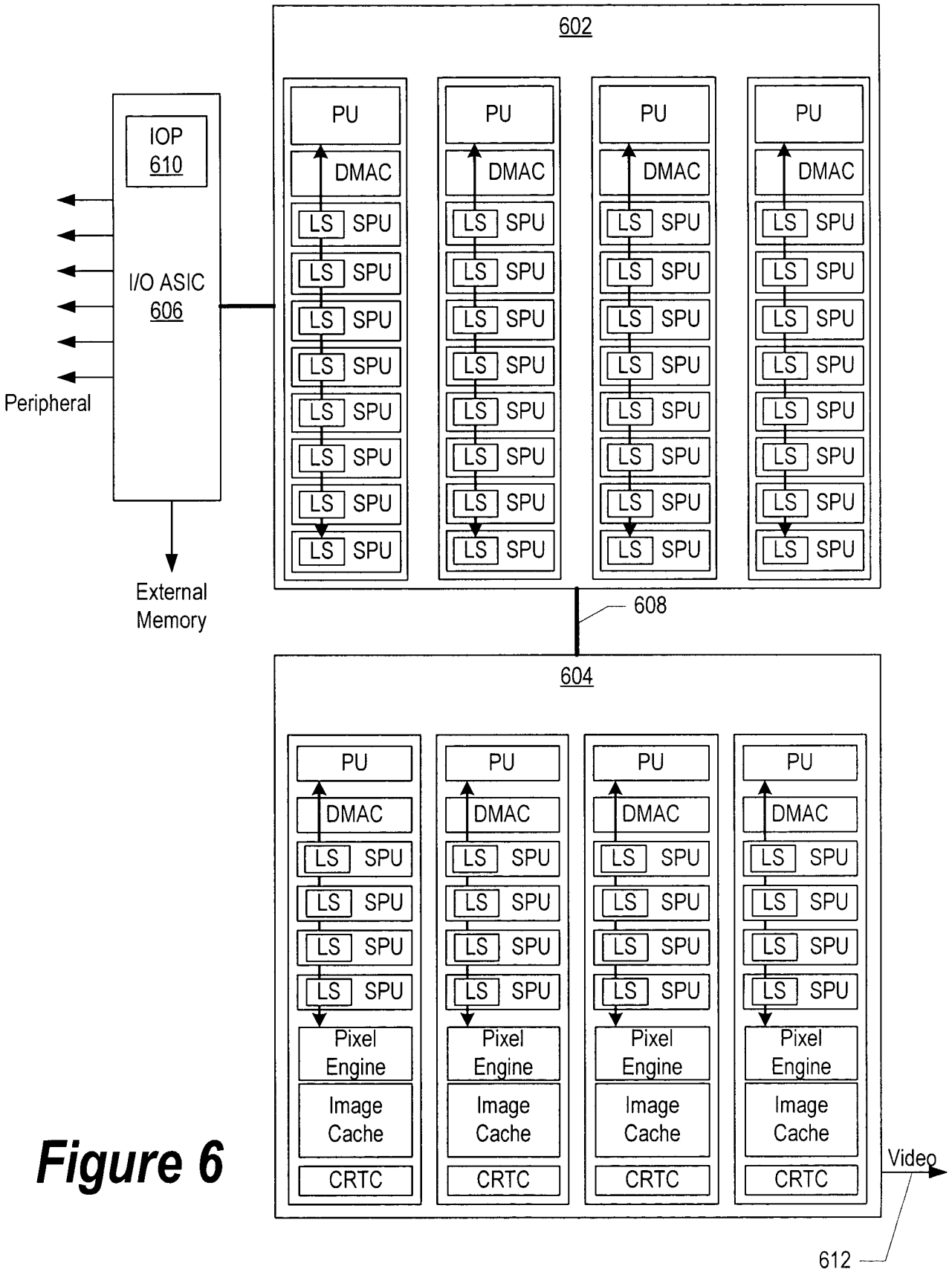
602
IOP
610
I/O ASIC
606
Peripheral
External
Memory
PU
DMAC
LS
SPU
608
604
Pixel
Engine
Image
Cache
CRTC
Video
612

Figure 6

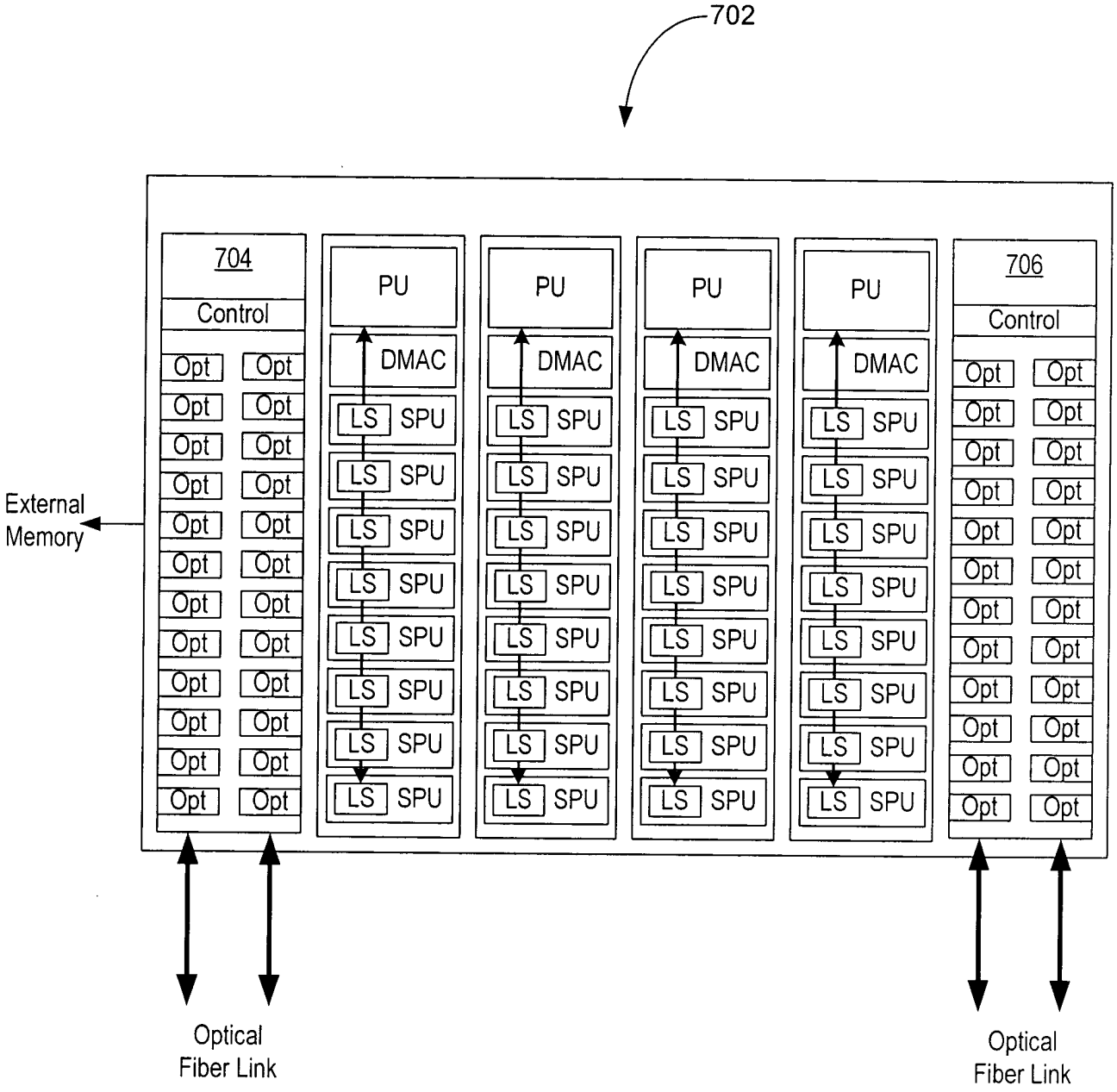
702
704
Control
Opt
PU
DMAC
LS
SPU
706
External Memory
Optical Fiber Link
Optical Fiber Link

Figure 7

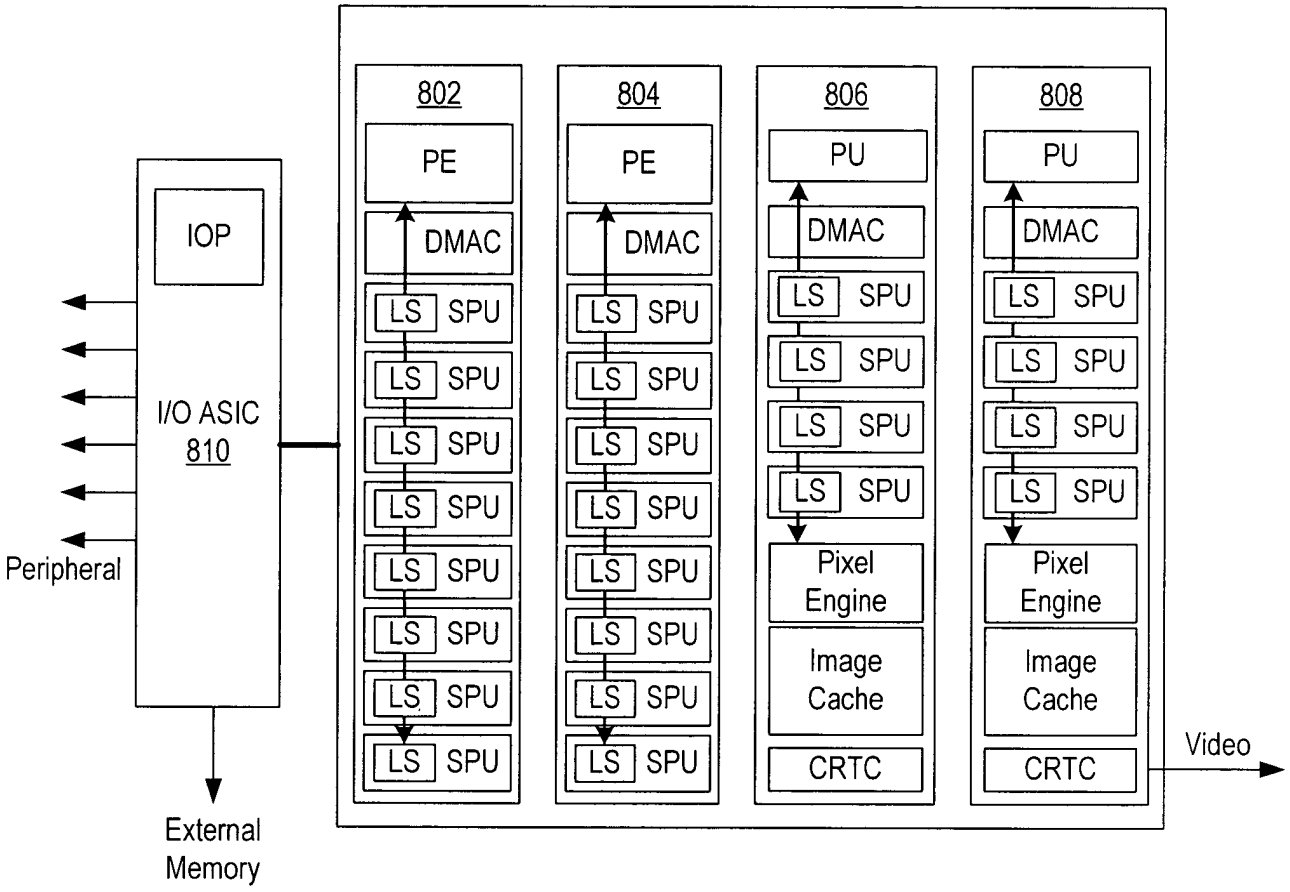
802
804
806
808
PE
PE
PU
PU
DMAC
DMAC
DMAC
DMAC
LS SPU
Pixel Engine
Image Cache
CRTC
IOP
I/O ASIC 810
Peripheral
External Memory
Video

Figure 8

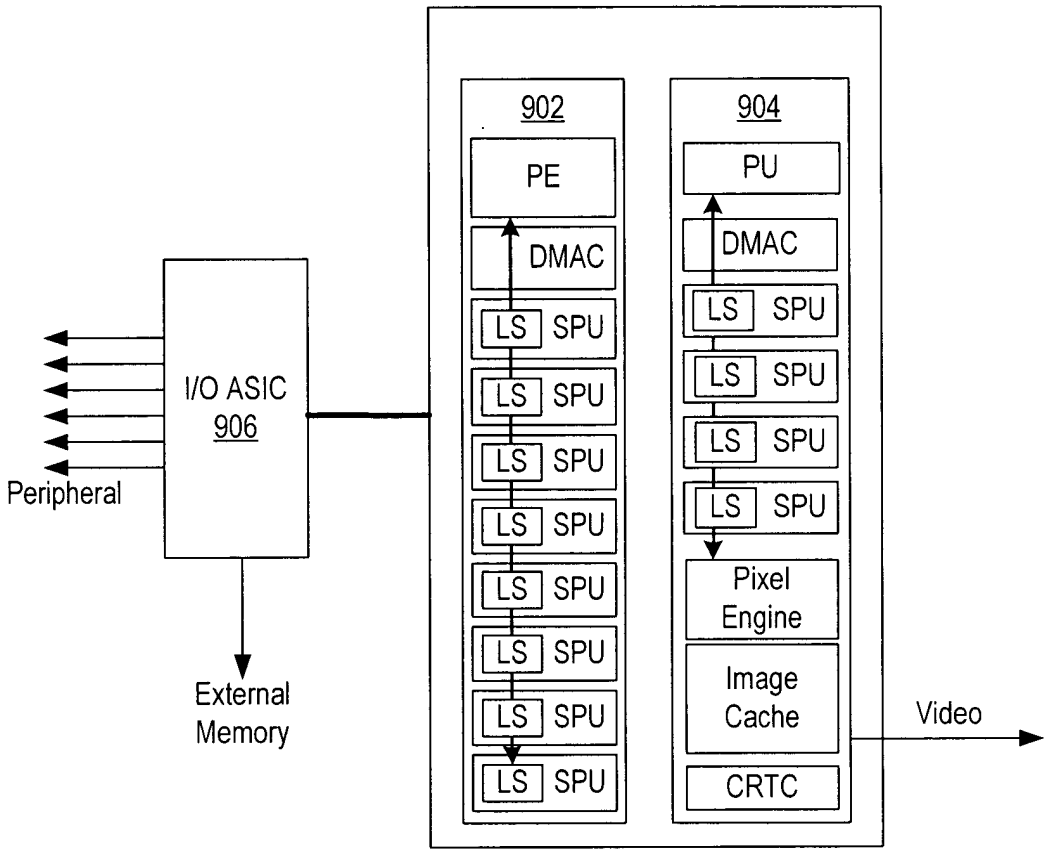
902
PE
DMAC
LS SPU
LS SPU
LS SPU
LS SPU
LS SPU
LS SPU
LS SPU
LS SPU
904
PU
DMAC
LS SPU
LS SPU
LS SPU
LS SPU
Pixel Engine
Image Cache
CRTC
Video
I/O ASIC 906
Peripheral
External Memory

Figure 9

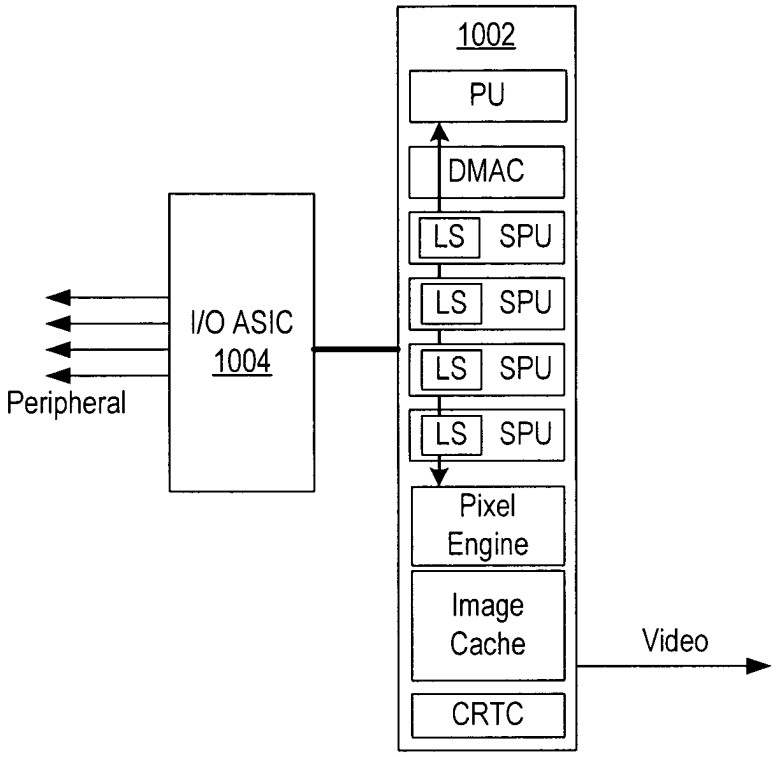
1002
PU
DMAC
LS
SPU
LS
SPU
LS
SPU
LS
SPU
I/O ASIC
1004
Peripheral
Pixel
Engine
Image
Cache
Video
CRTC

Figure 10

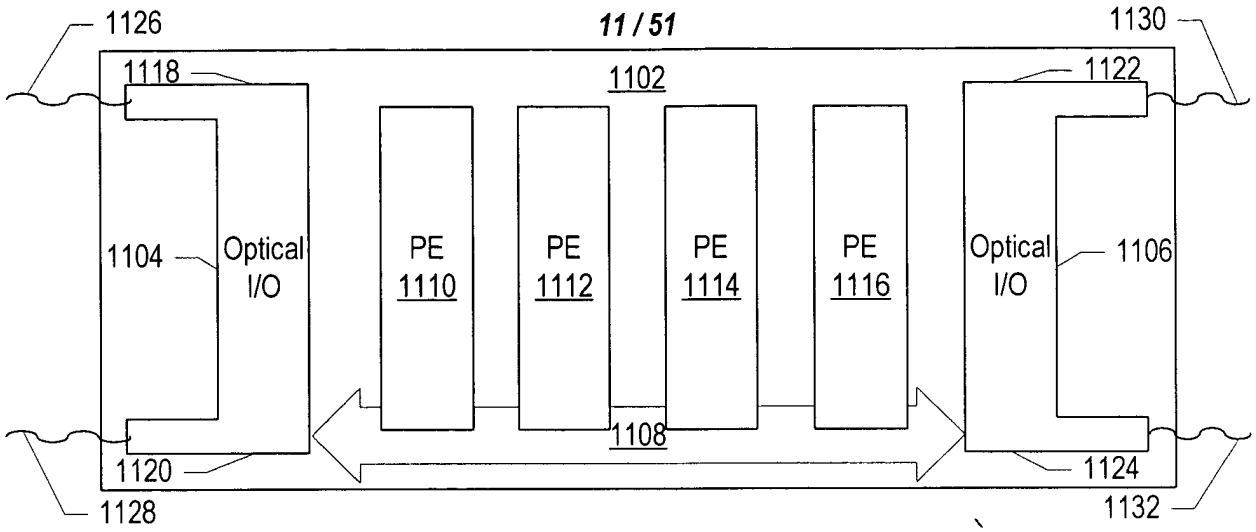

Figure 11A

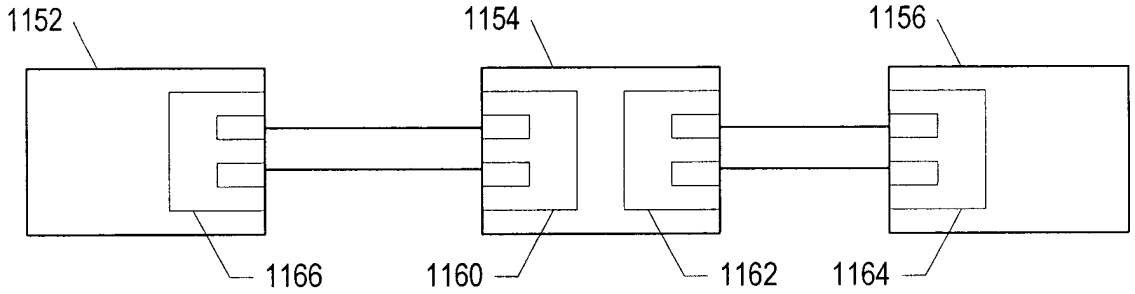

Figure 11B

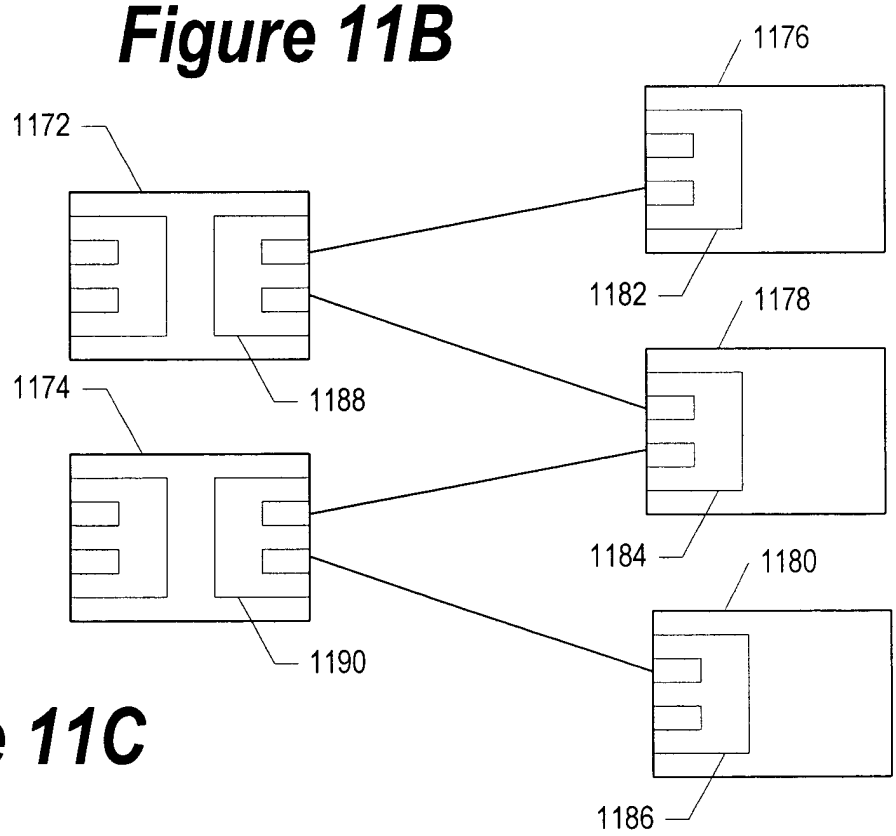

Figure 11C

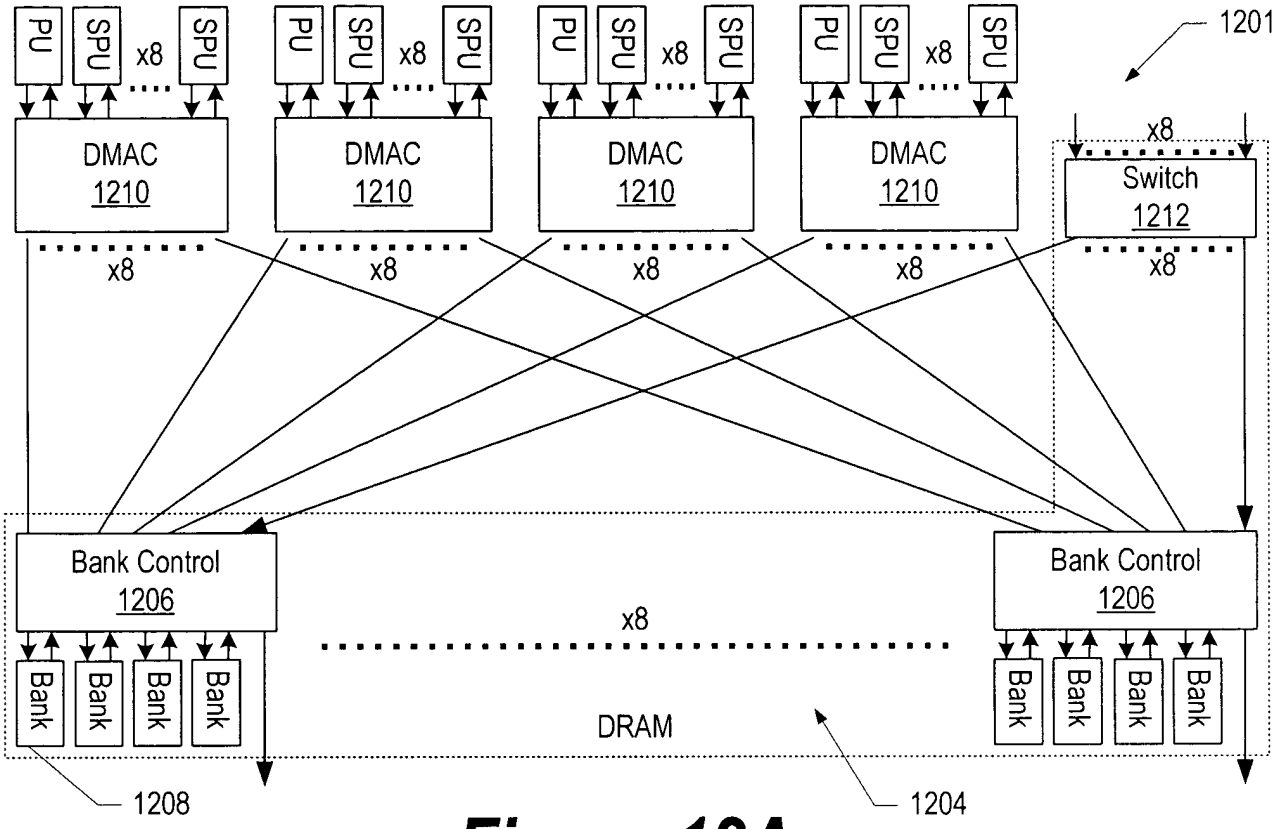

Figure 12A

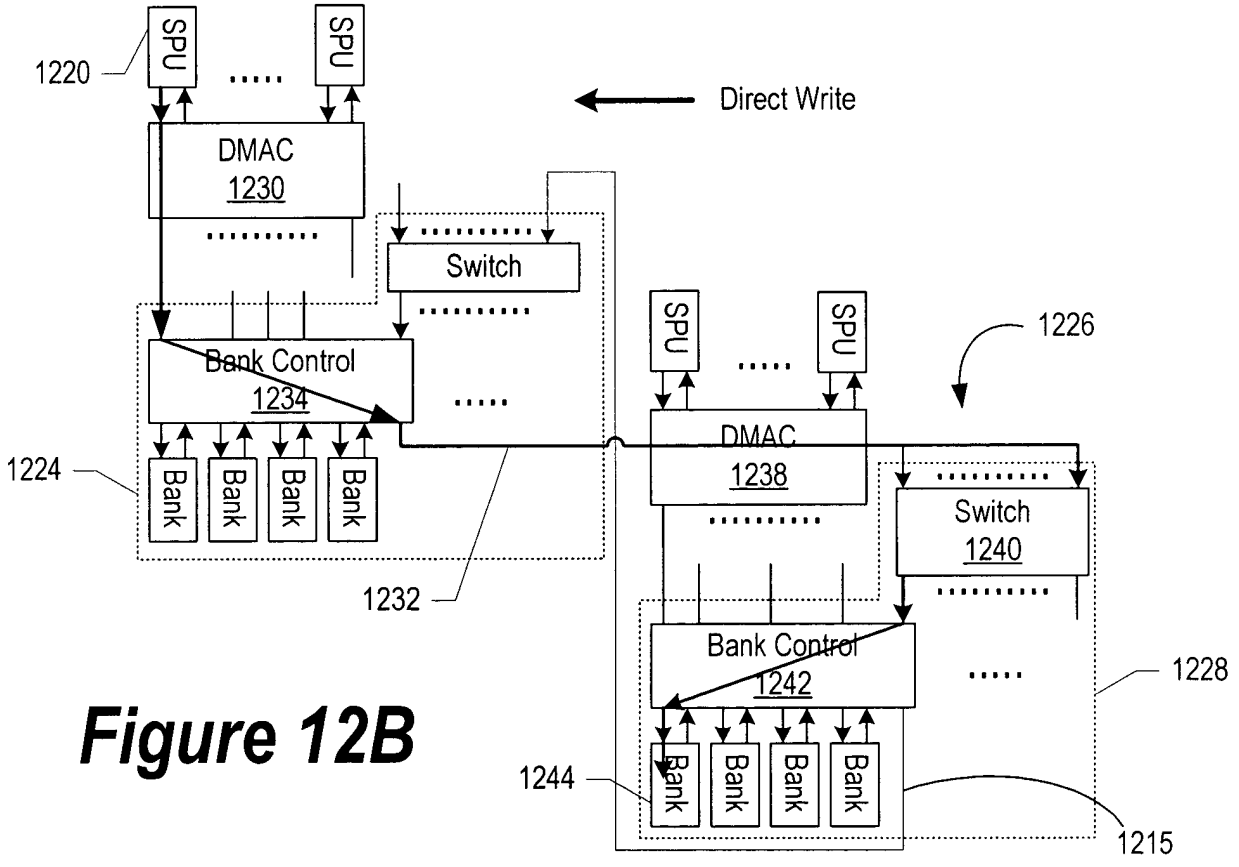

Figure 12B

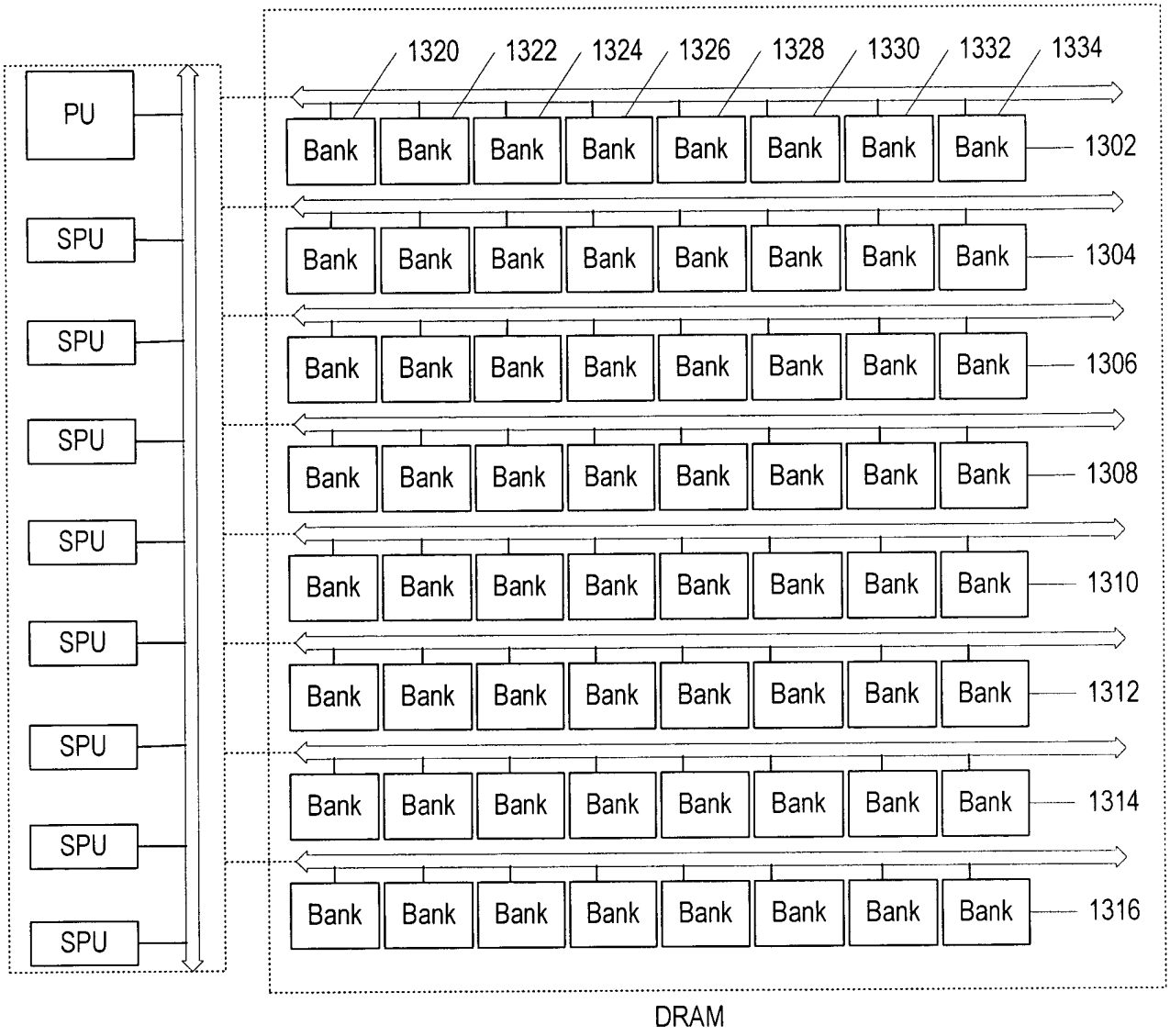
1320
1322
1324
1326
1328
1330
1332
1334
PU
SPU
SPU
SPU
SPU
SPU
SPU
SPU
SPU
Bank
1302
1304
1306
1308
1310
1312
1314
1316
DRAM

Figure 13

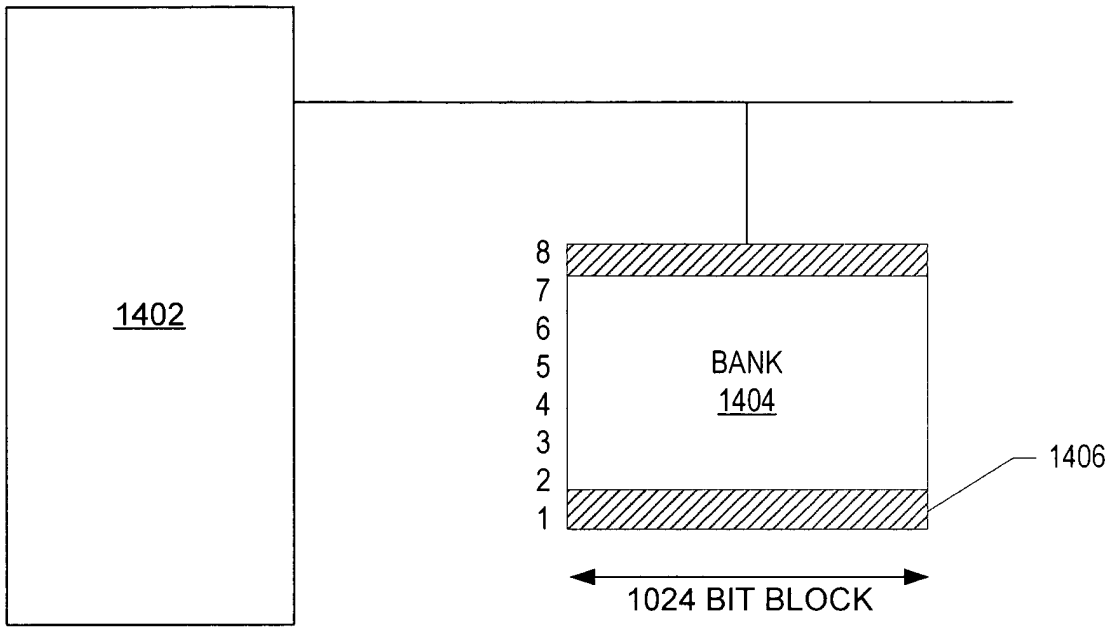

Figure 14A

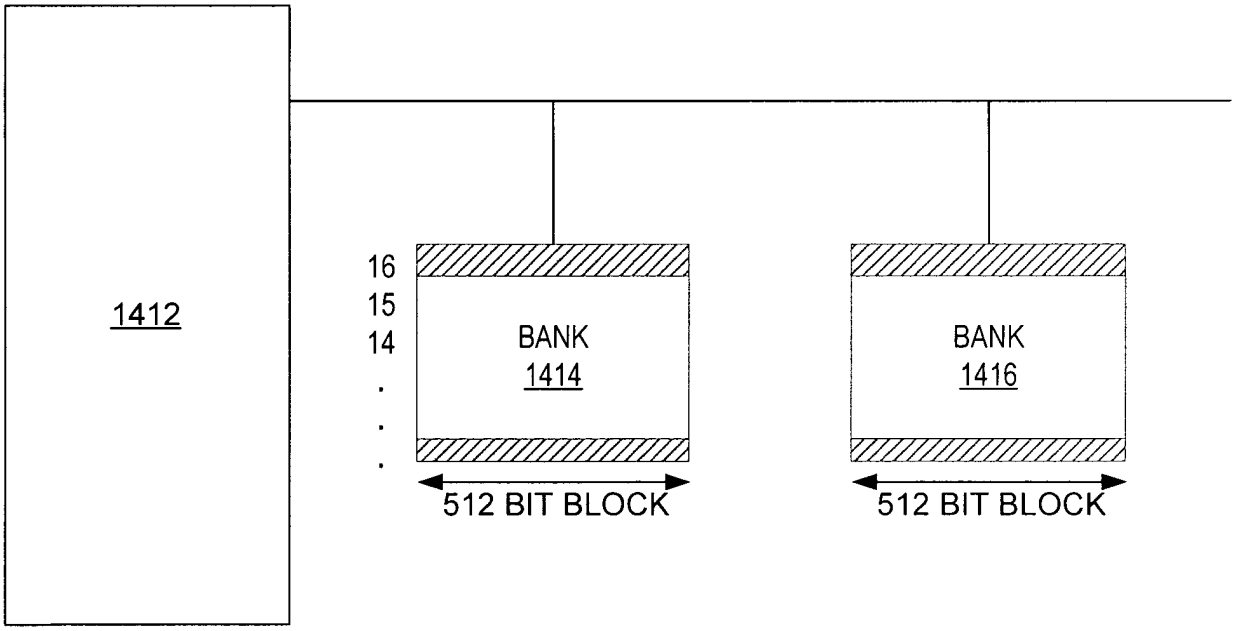

Figure 14B

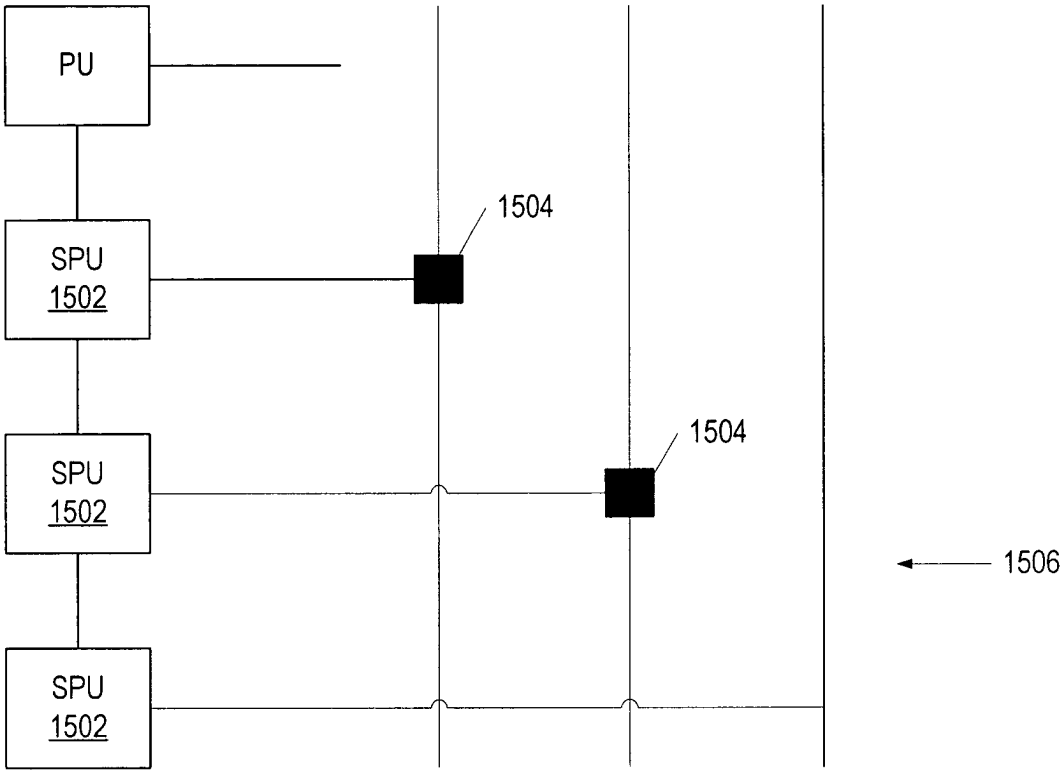

Figure 15

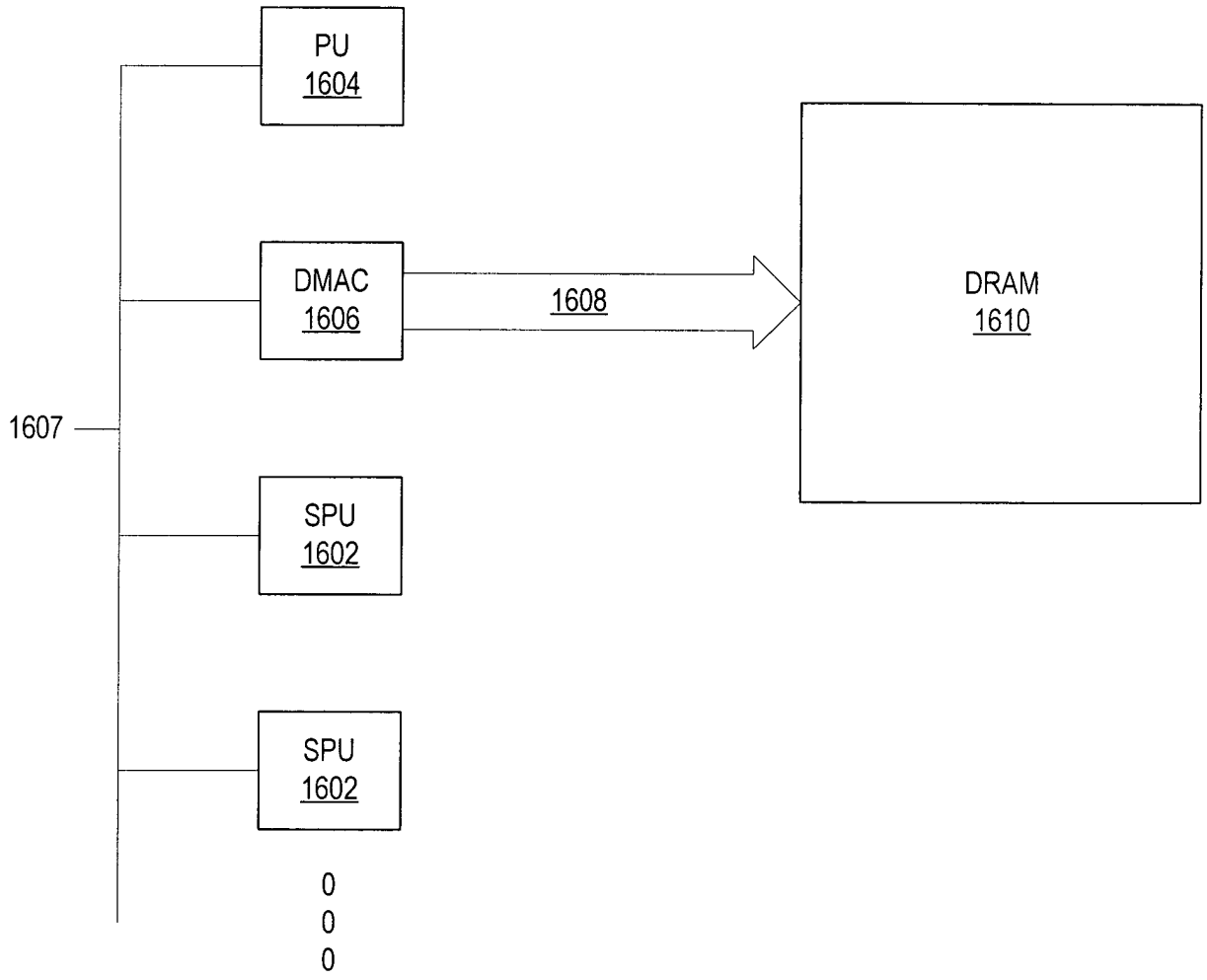

Figure 16

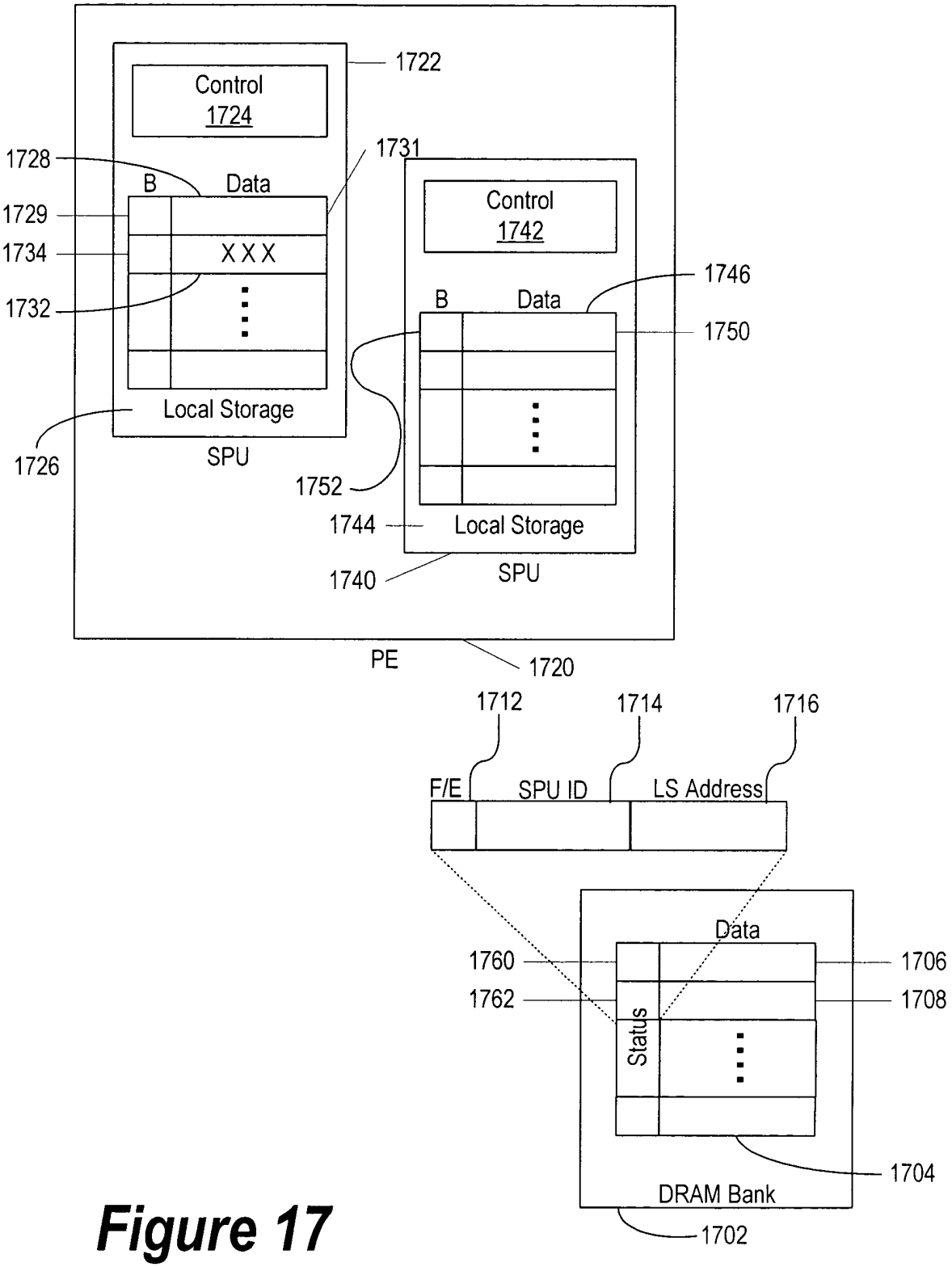
1722
Control
1724
1728
B
Data
1731
1729
1734
XXX
1732
Control
1742
1746
B
Data
1750
Local Storage
1726
SPU
1752
1744
Local Storage
1740
SPU
PE
1720
1712
1714
1716
F/E
SPU ID
LS Address
Data
1760
1706
1762
1708
Status
1704
DRAM Bank
1702

Figure 17

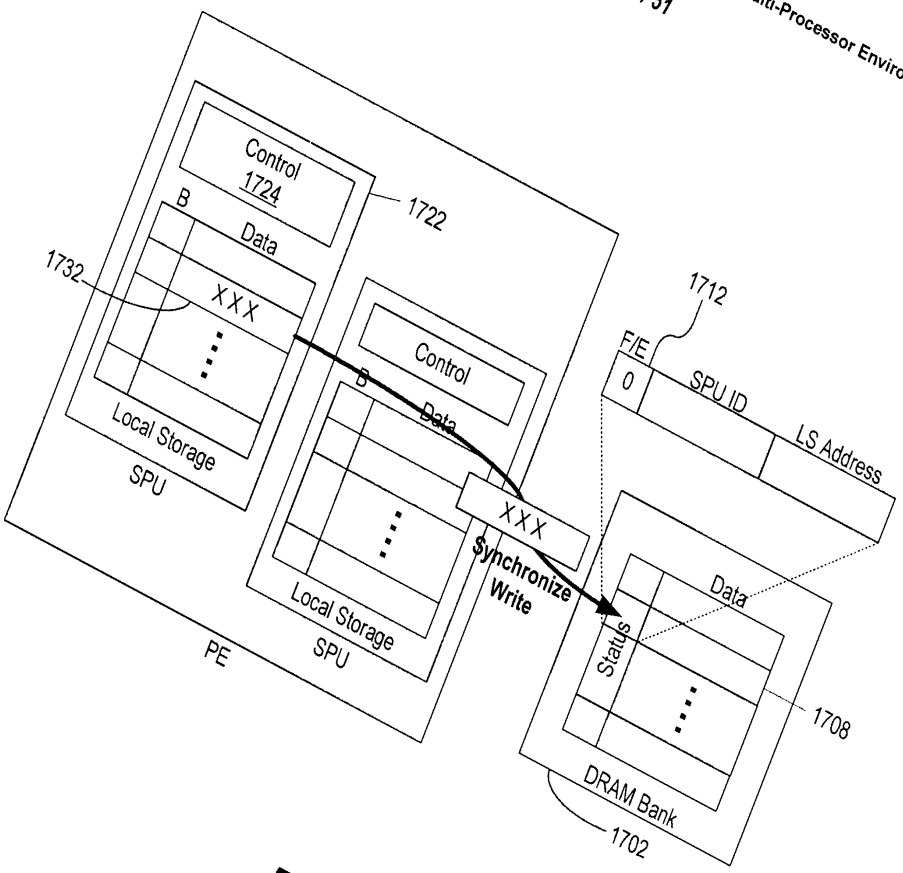
Control
1724
B
Data
1722
1732
XXX
Local Storage
SPU
Control
B
Data
XXX
Local Storage
SPU
PE
1712
F/E
0
SPU ID
LS Address
Synchronize
Write
Data
Status
1708
DRAM Bank
1702

Figure 18

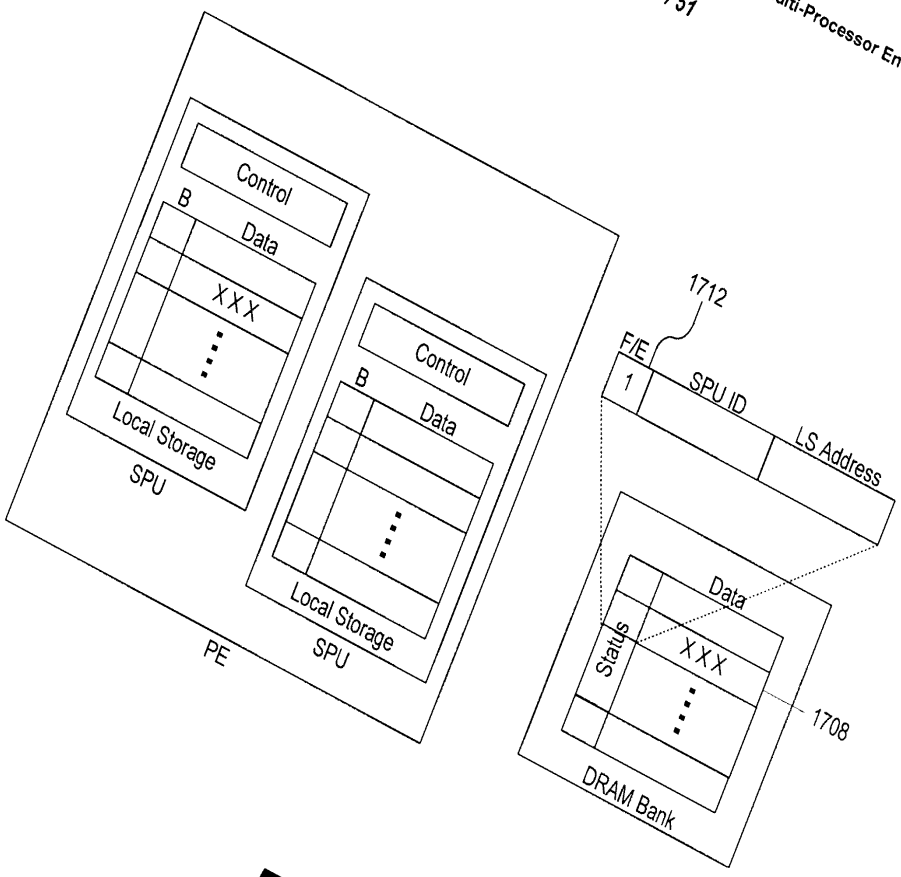
Control
B
Data
XXX
Local Storage
SPU
Control
B
Data
Local Storage
SPU
PE
1712
F/E
1
SPU ID
LS Address
Data
Status
XXX
1708
DRAM Bank

Figure 19

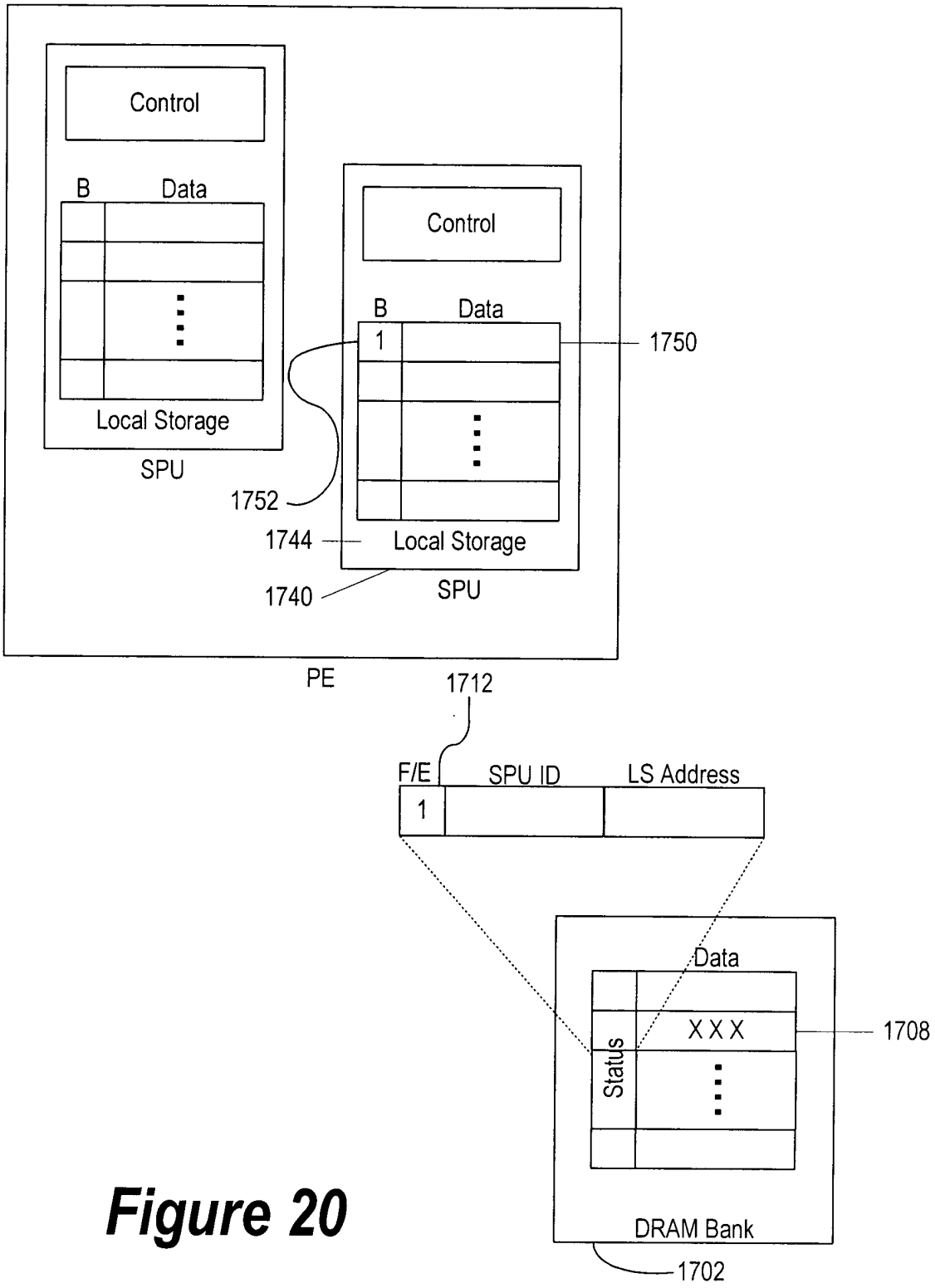
Control
B
Data
Local Storage
SPU
Control
B
Data
1
1750
1752
1744
Local Storage
1740
SPU
PE
1712
F/E
SPU ID
LS Address
1
Data
XXX
Status
1708
DRAM Bank
1702

Figure 20

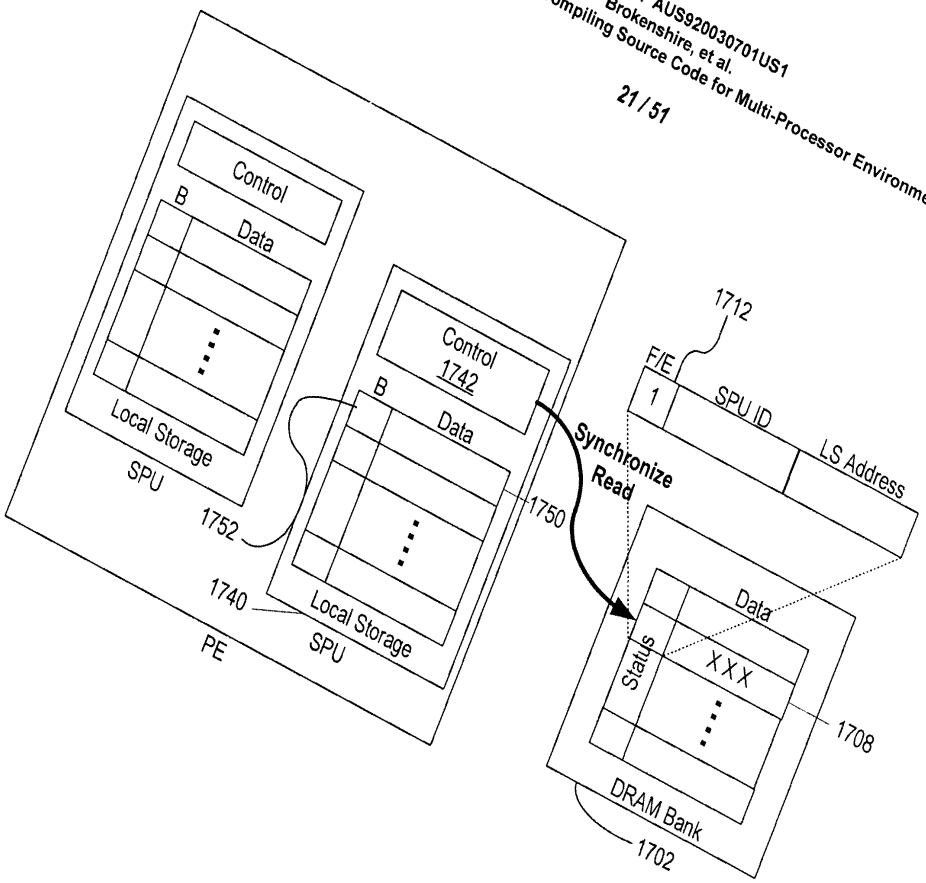
Control
B
Data
Local Storage
SPU
1752
Control
1742
B
Data
1750
1740
Local Storage
SPU
PE
1712
F/E
1
SPU ID
LS Address
Synchronize
Read
Data
Status
XXX
1708
DRAM Bank
1702

Figure 21

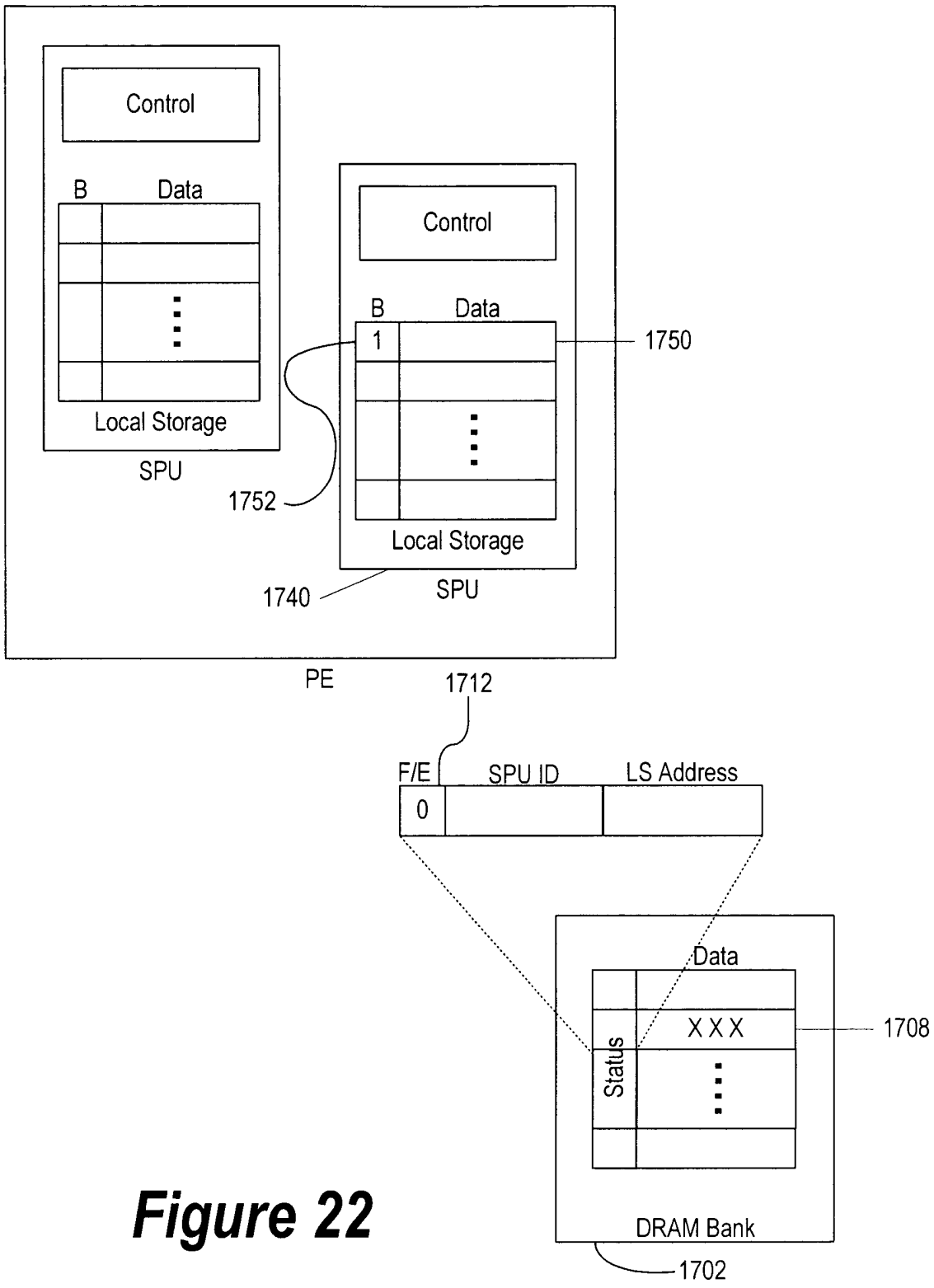
Control
B
Data
Local Storage
SPU
Control
B
Data
1
1750
1752
Local Storage
SPU
1740
PE
1712
F/E
SPU ID
LS Address
0
Data
XXX
Status
1708
DRAM Bank
1702

Figure 22

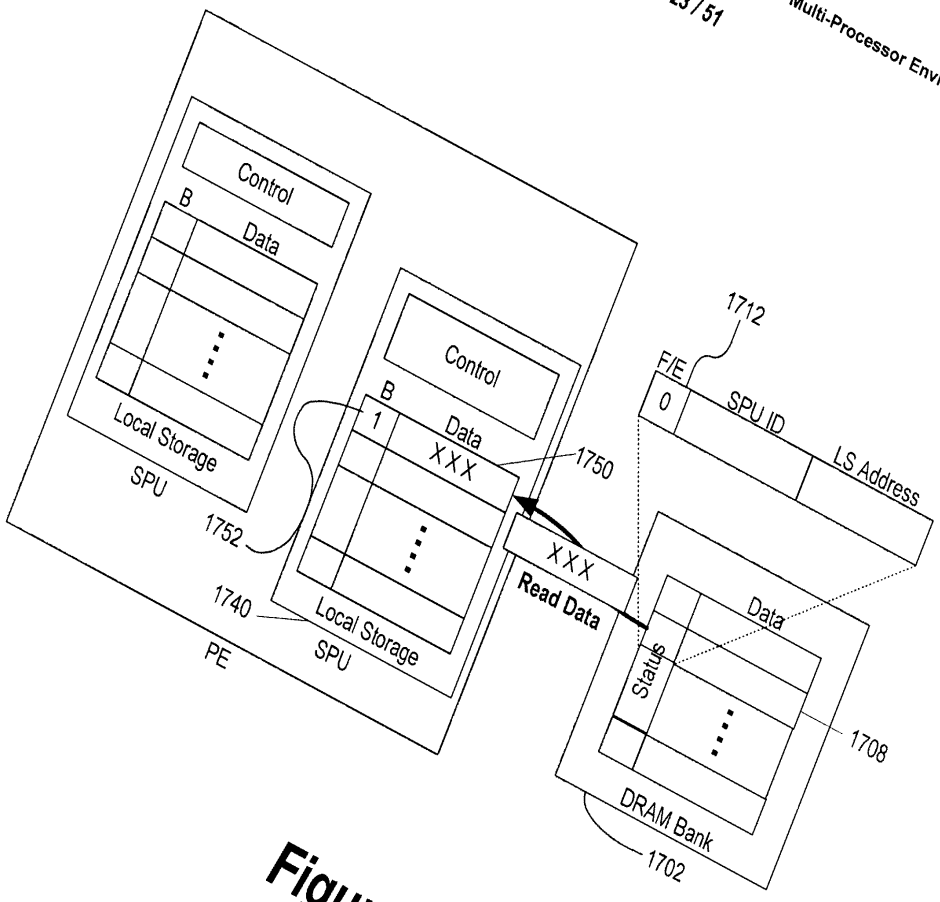
Control
B
Data
Local Storage
SPU
1752
Control
B
1
Data
XXX
1750
1740
Local Storage
SPU
PE
1712
F/E
0
SPU ID
LS Address
XXX
Read Data
Data
Status
1708
DRAM Bank
1702

Figure 23

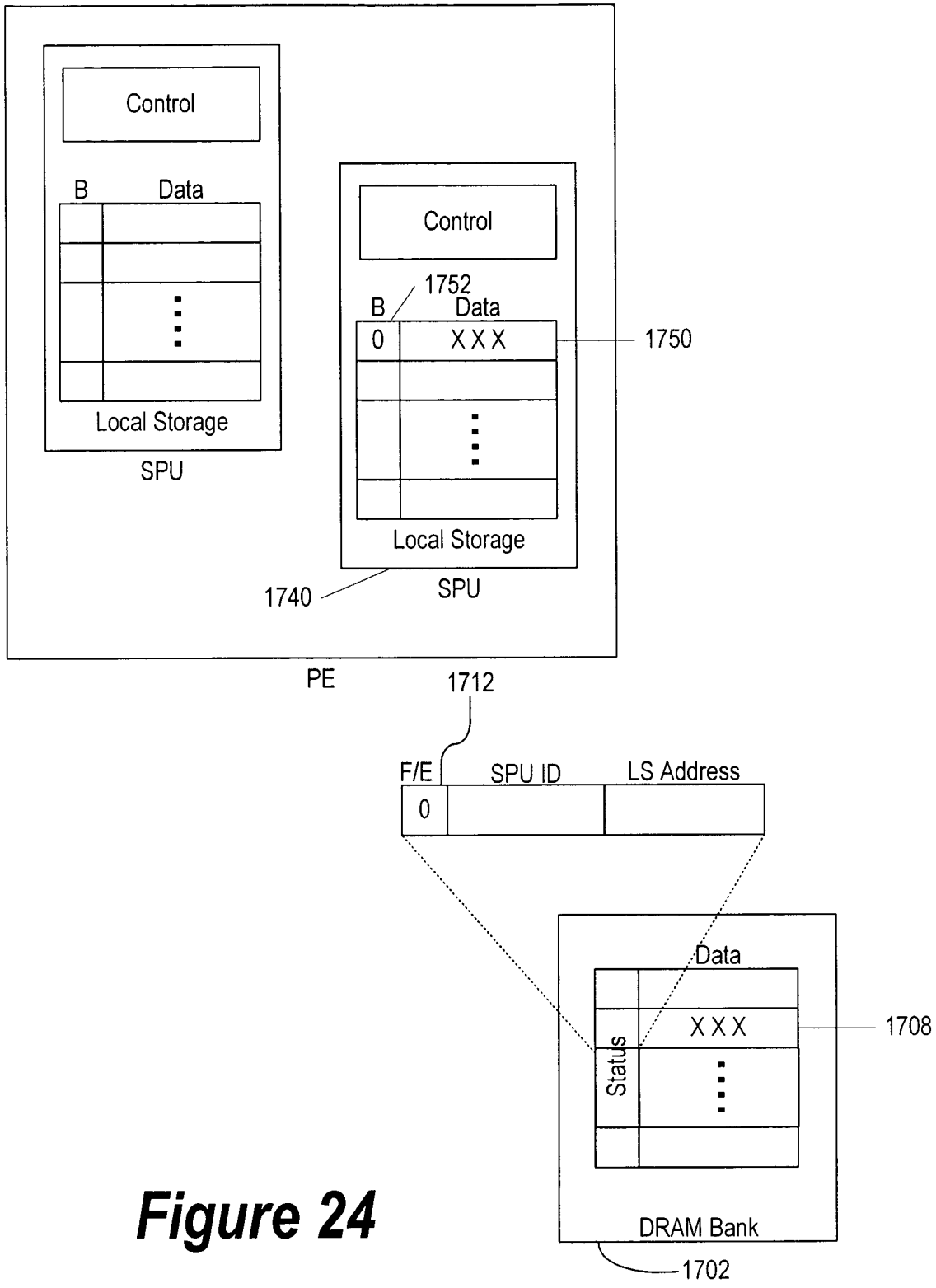
Control
B
Data
Local Storage
SPU
Control
1752
B
Data
0
X X X
1750
Local Storage
SPU
1740
PE
1712
F/E
SPU ID
LS Address
0
Data
X X X
1708
Status
DRAM Bank
1702

Figure 24

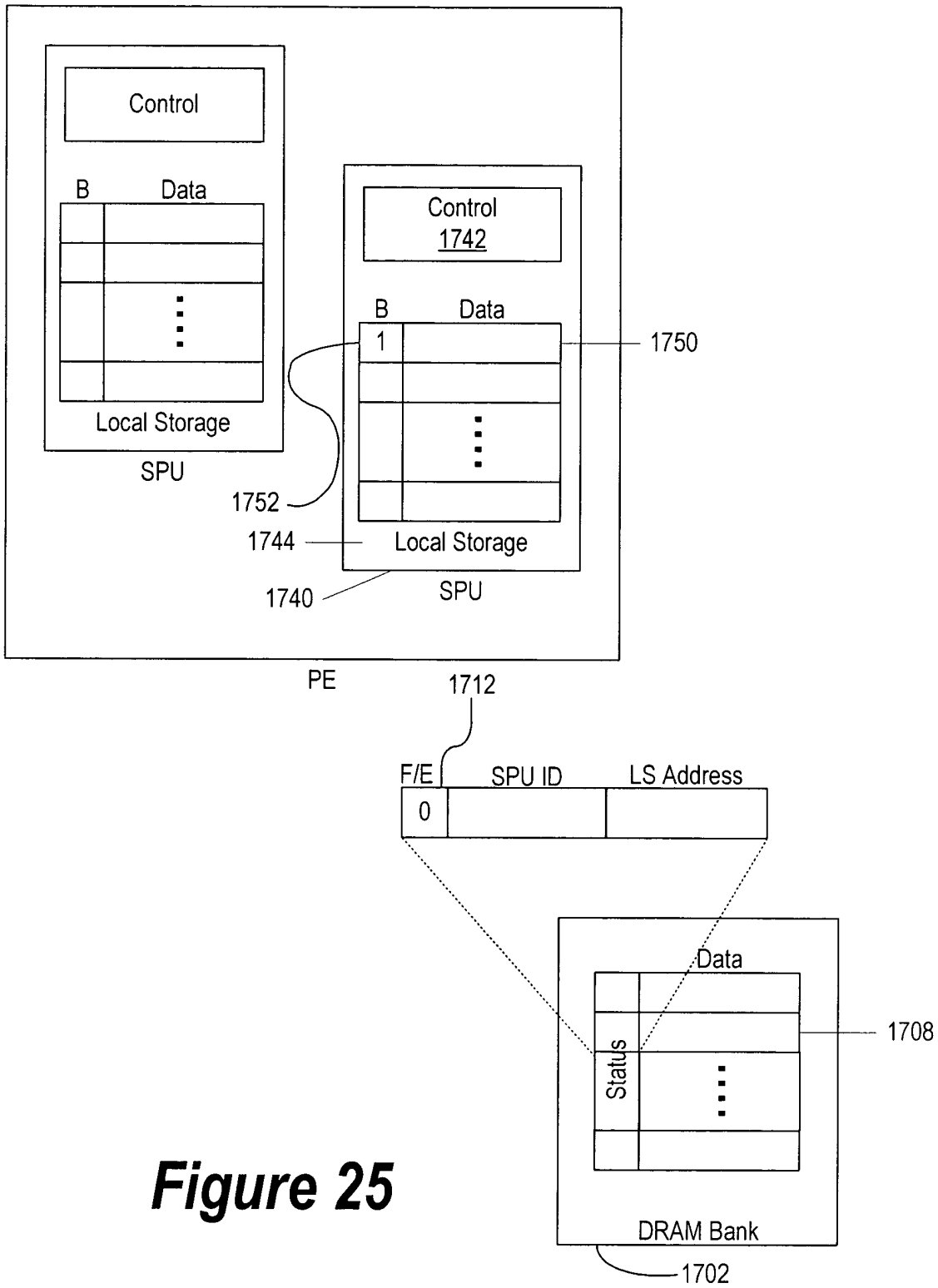
Control
B
Data
Local Storage
SPU
Control
1742
B
Data
1
1750
1752
1744
Local Storage
1740
SPU
PE
1712
F/E
SPU ID
LS Address
0
Data
1708
Status
DRAM Bank
1702

Figure 25

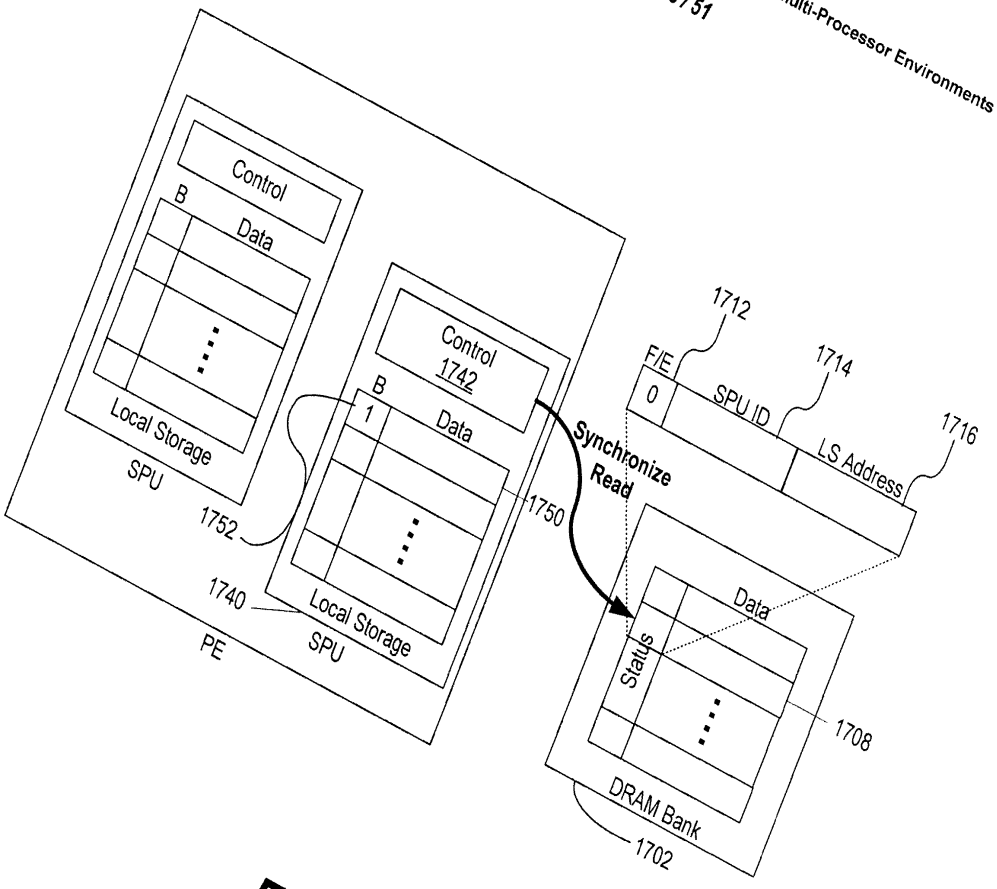
Control
B
Data
Local Storage
SPU
1752
Control
1742
B
1
Data
1750
1740
Local Storage
SPU
PE
1712
F/E
0
SPU ID
1714
LS Address
1716
Synchronize
Read
Data
Status
1708
DRAM Bank
1702

Figure 26

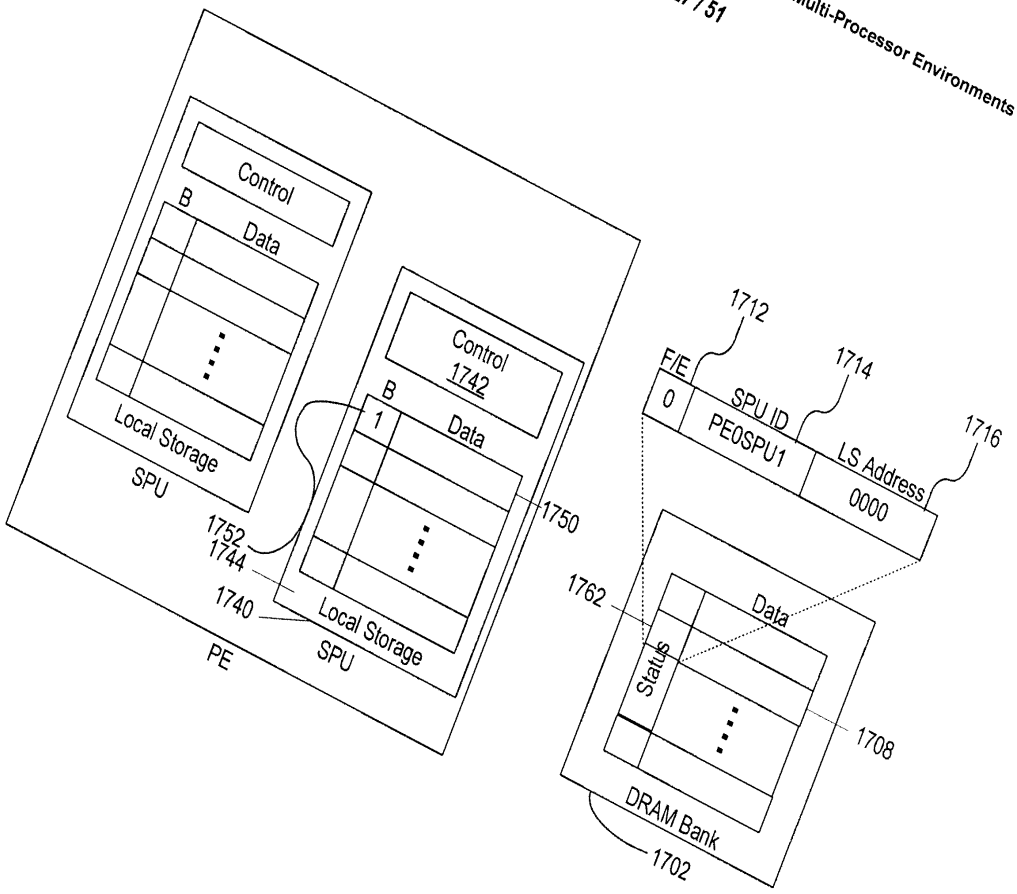
Control
B
Data
Local Storage
SPU
Control
1742
B
1
Data
1750
1752
1744
1740
Local Storage
SPU
PE
1712
F/E
0
1714
SPU ID
PE0SPU1
LS Address
0000
1716
1762
Data
Status
1708
DRAM Bank
1702

Figure 27

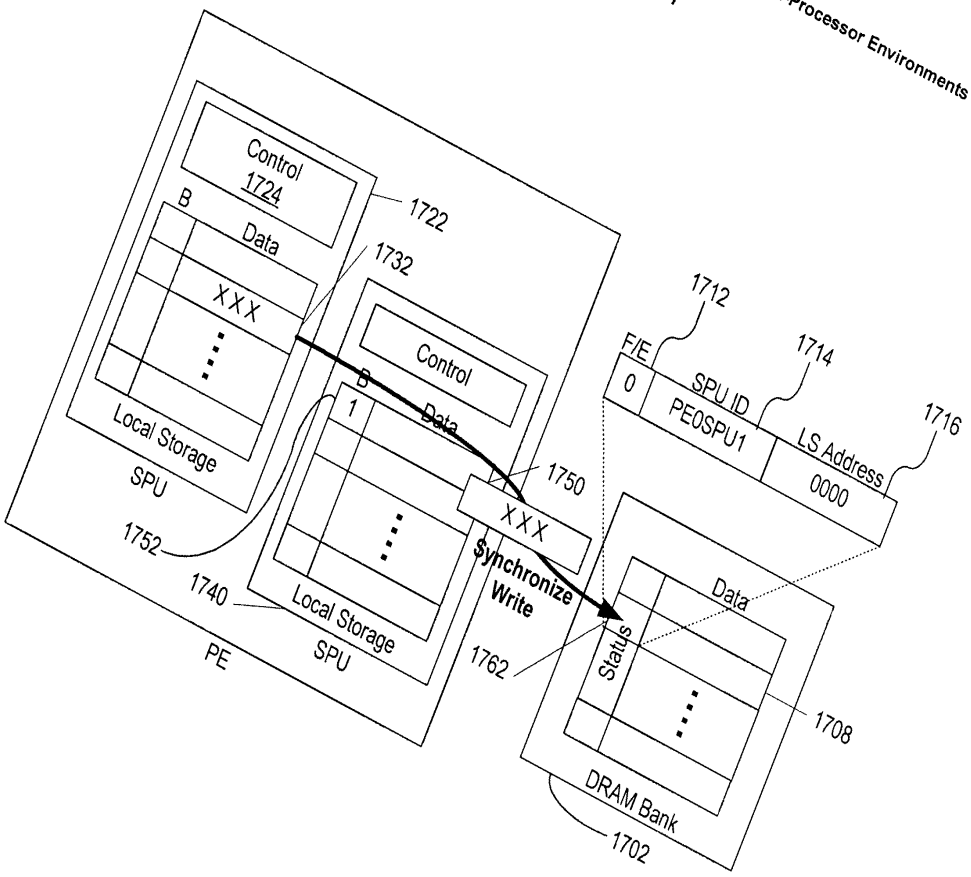
Control
1724
B
Data
XXX
1722
1732
Local Storage
SPU
1752
1740
PE
Control
B
1
Data
Local Storage
SPU
1750
XXX
Synchronize
Write
1762
1712
F/E
0
SPU ID
PE0SPU1
1714
LS Address
0000
1716
Data
Status
1708
DRAM Bank
1702

Figure 28

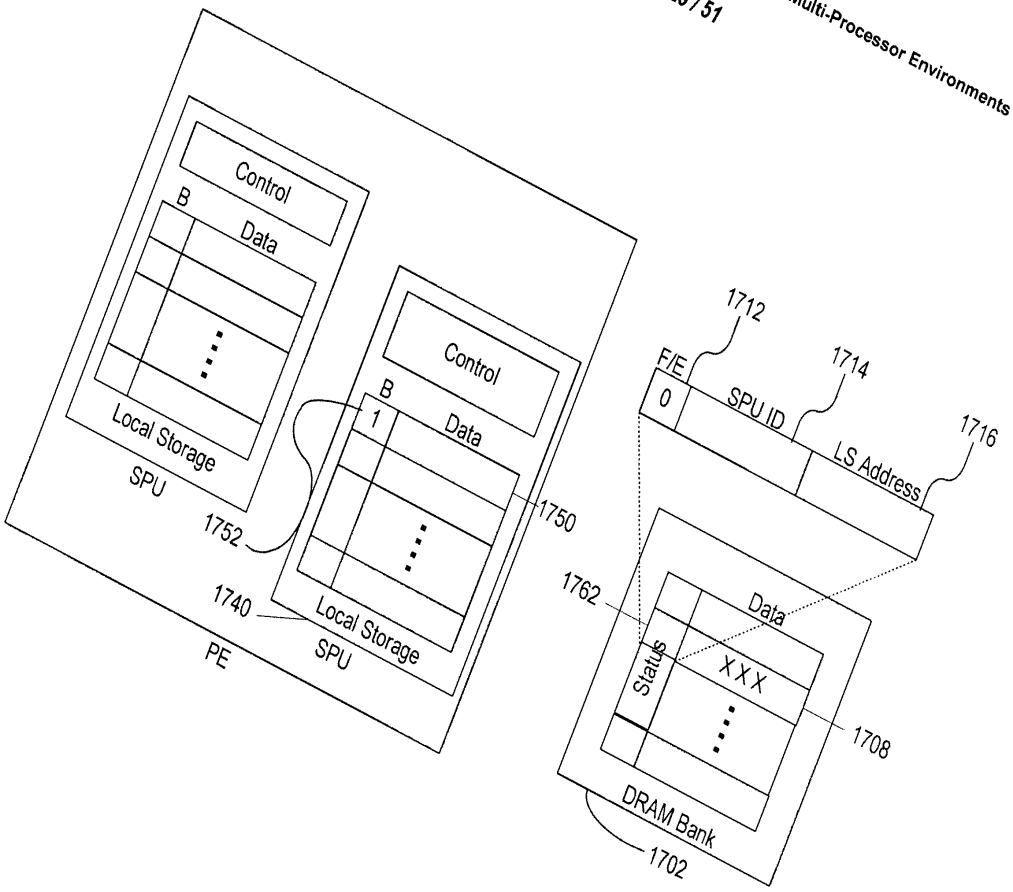
Control
B
Data
Local Storage
SPU
1752
Control
B
1
Data
1750
1740
Local Storage
SPU
PE
1712
F/E
0
1714
SPU ID
1716
LS Address
1762
Data
Status
XXX
1708
DRAM Bank
1702

Figure 29

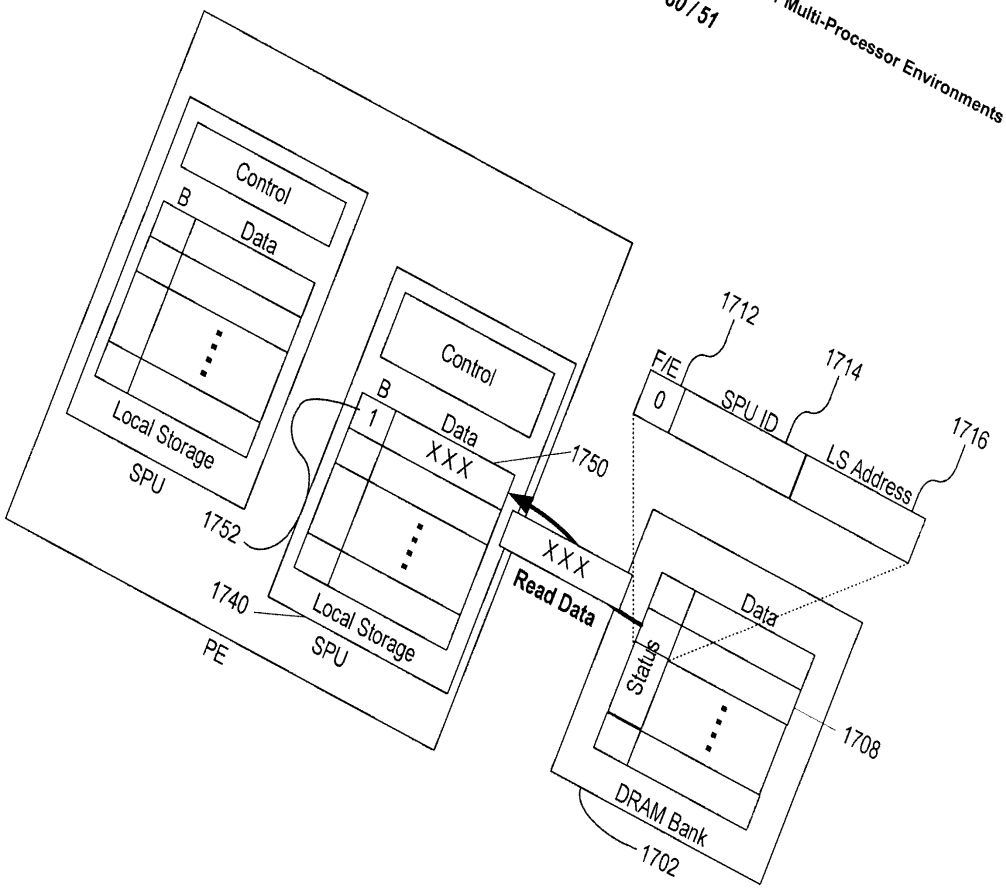
Control
B
Data
Local Storage
SPU
1752
Control
B
1
Data
XXX
1750
1740
Local Storage
SPU
PE
1712
F/E
0
1714
SPU ID
1716
LS Address
XXX
Read Data
Data
Status
1708
DRAM Bank
1702

Figure 30

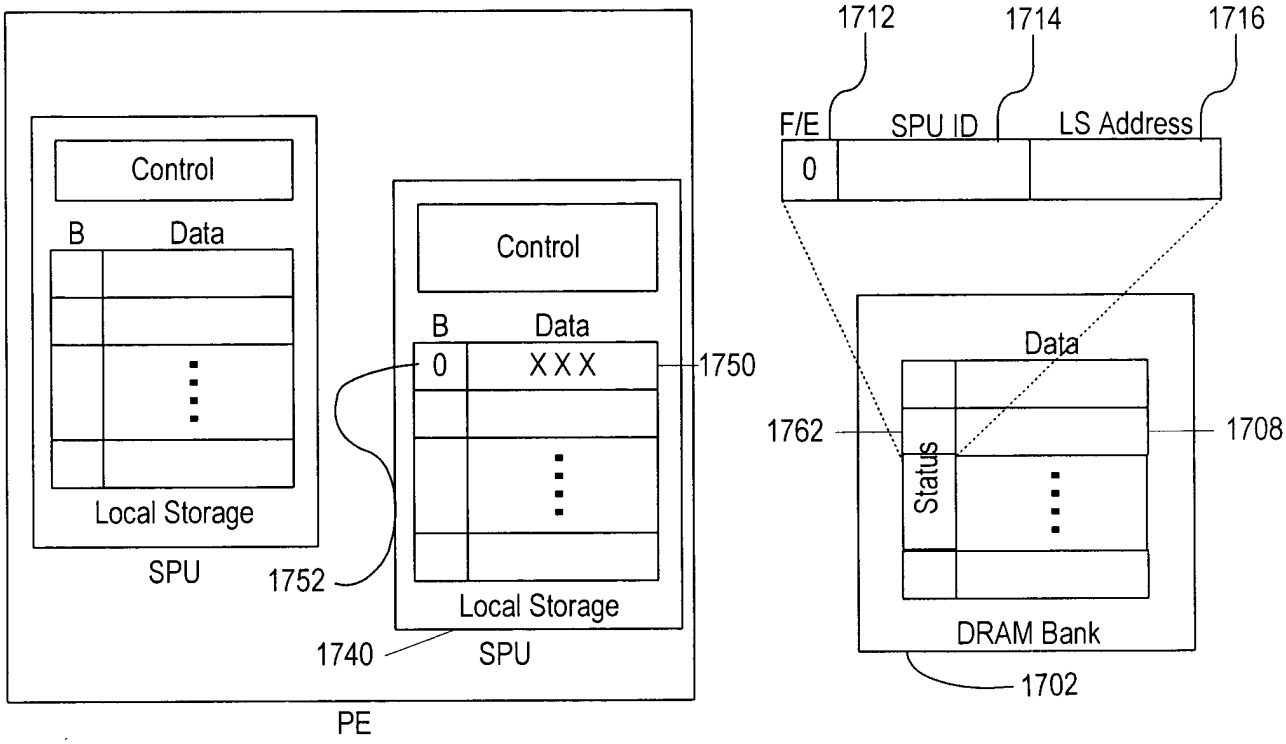
1712
1714
1716
F/E
SPU ID
LS Address
0
Control
B
Data
Local Storage
SPU
1752
Control
B
Data
0
XXX
1750
Local Storage
1740
SPU
PE
Data
1762
1708
Status
DRAM Bank
1702

Figure 31

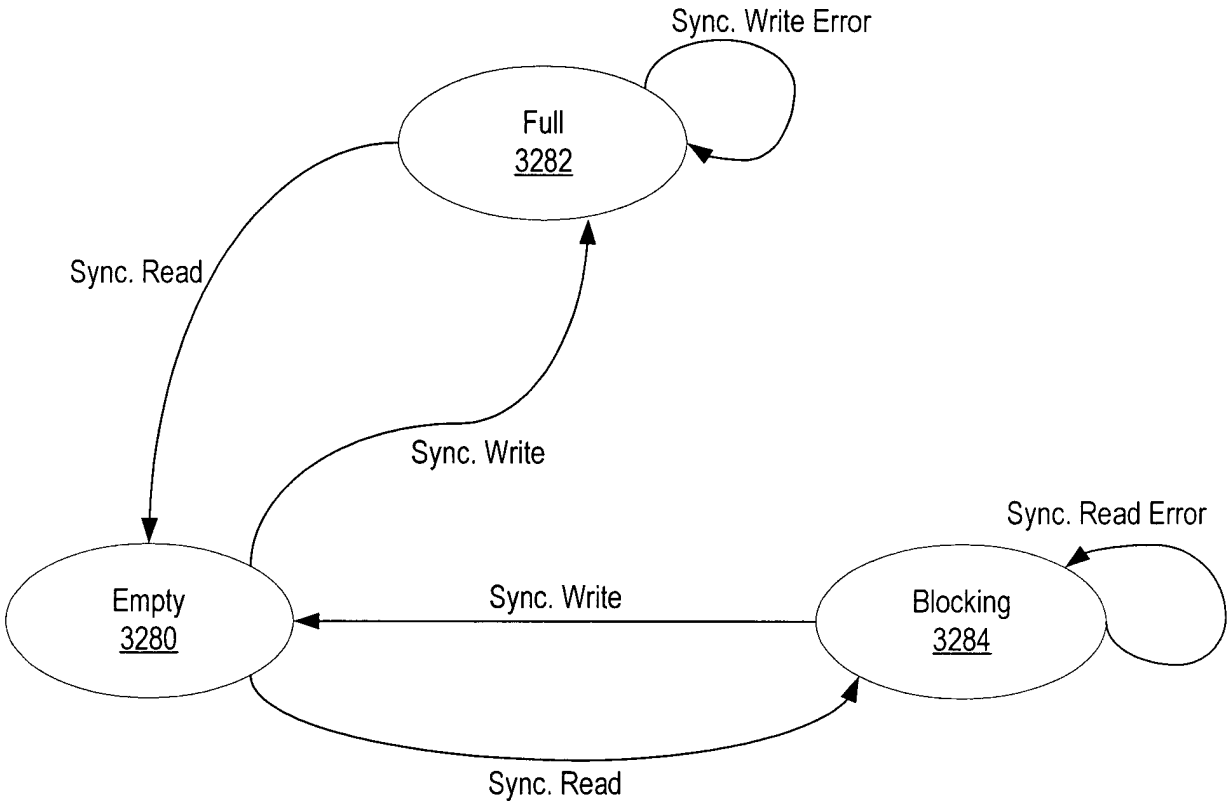
Sync. Write Error
Full
3282
Sync. Read
Sync. Write
Sync. Read Error
Empty
3280
Sync. Write
Blocking
3284
Sync. Read

Figure 32

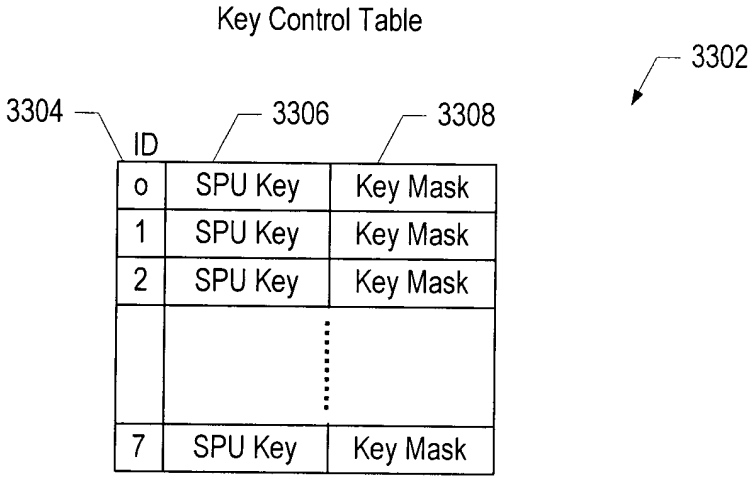
Key Control Table
3302
3304
3306
3308
ID
0
SPU Key
Key Mask
1
SPU Key
Key Mask
2
SPU Key
Key Mask
7
SPU Key
Key Mask

Figure 33

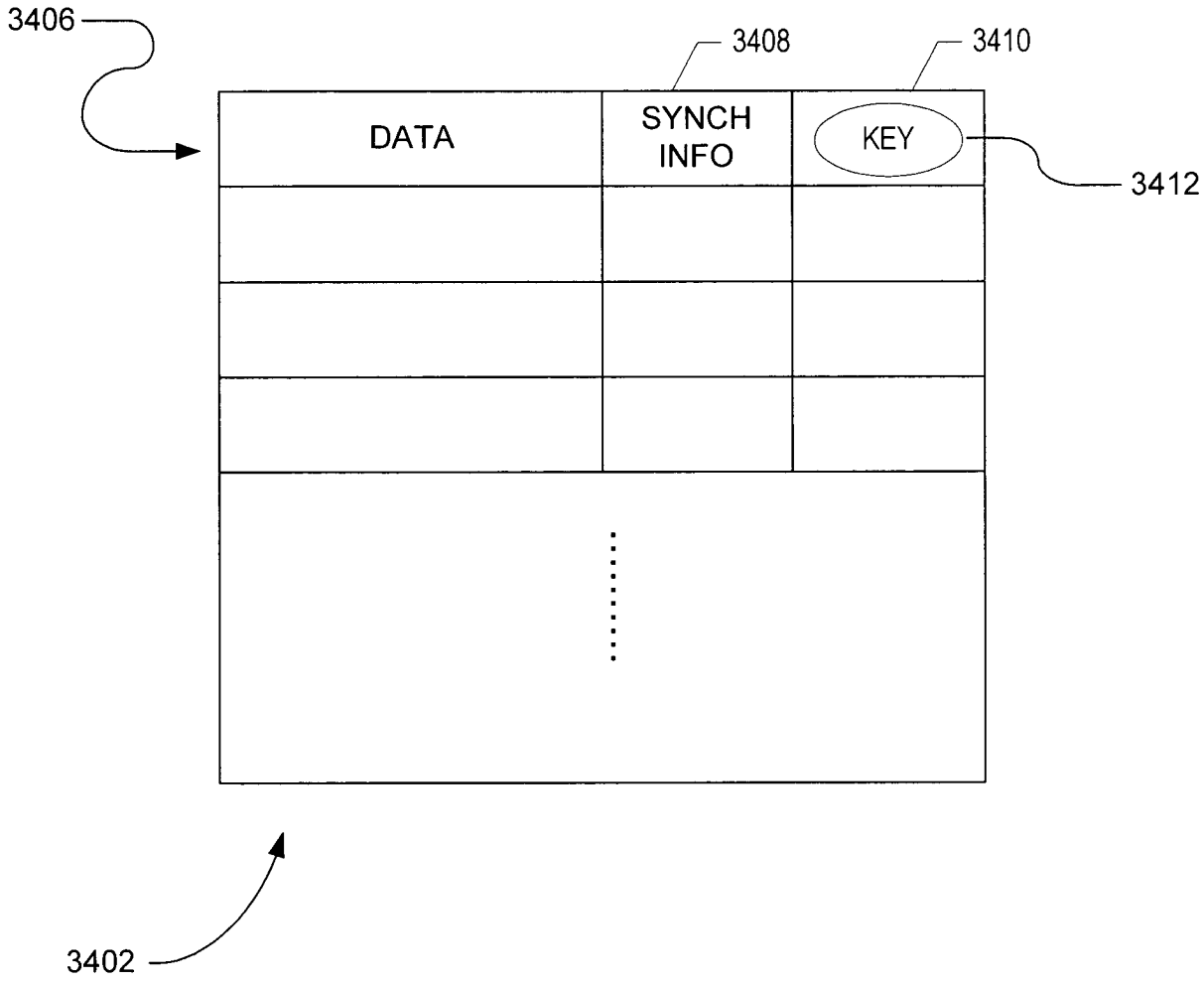

Figure 34

Memory Access Control Table

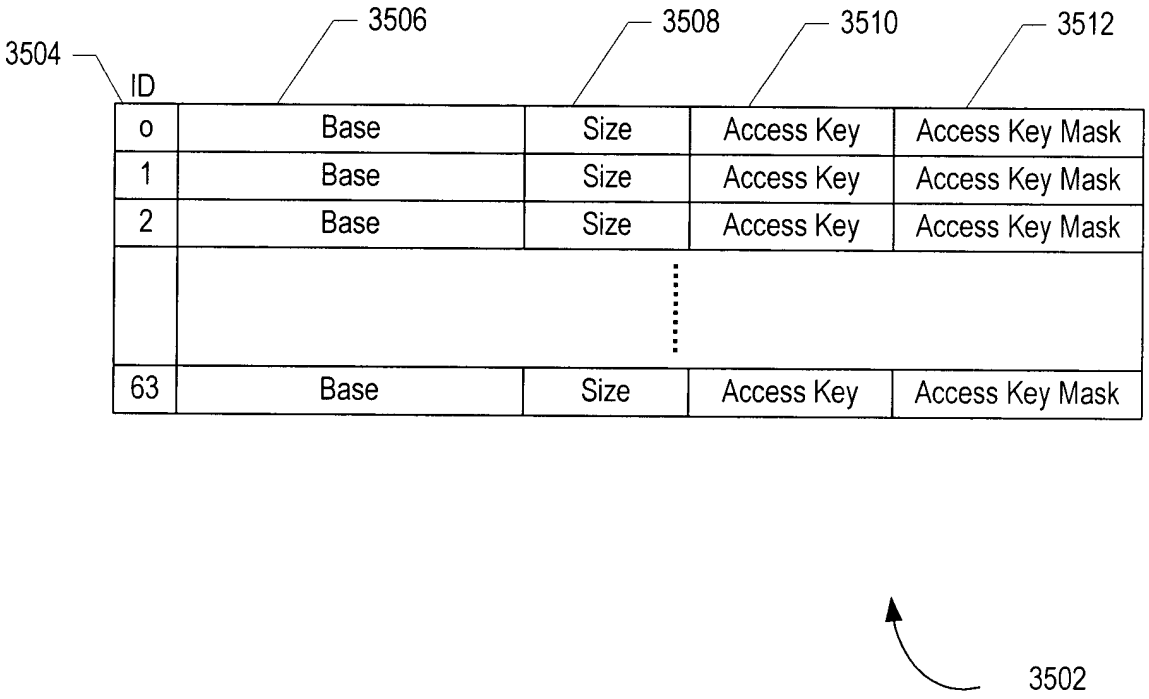

Figure 35

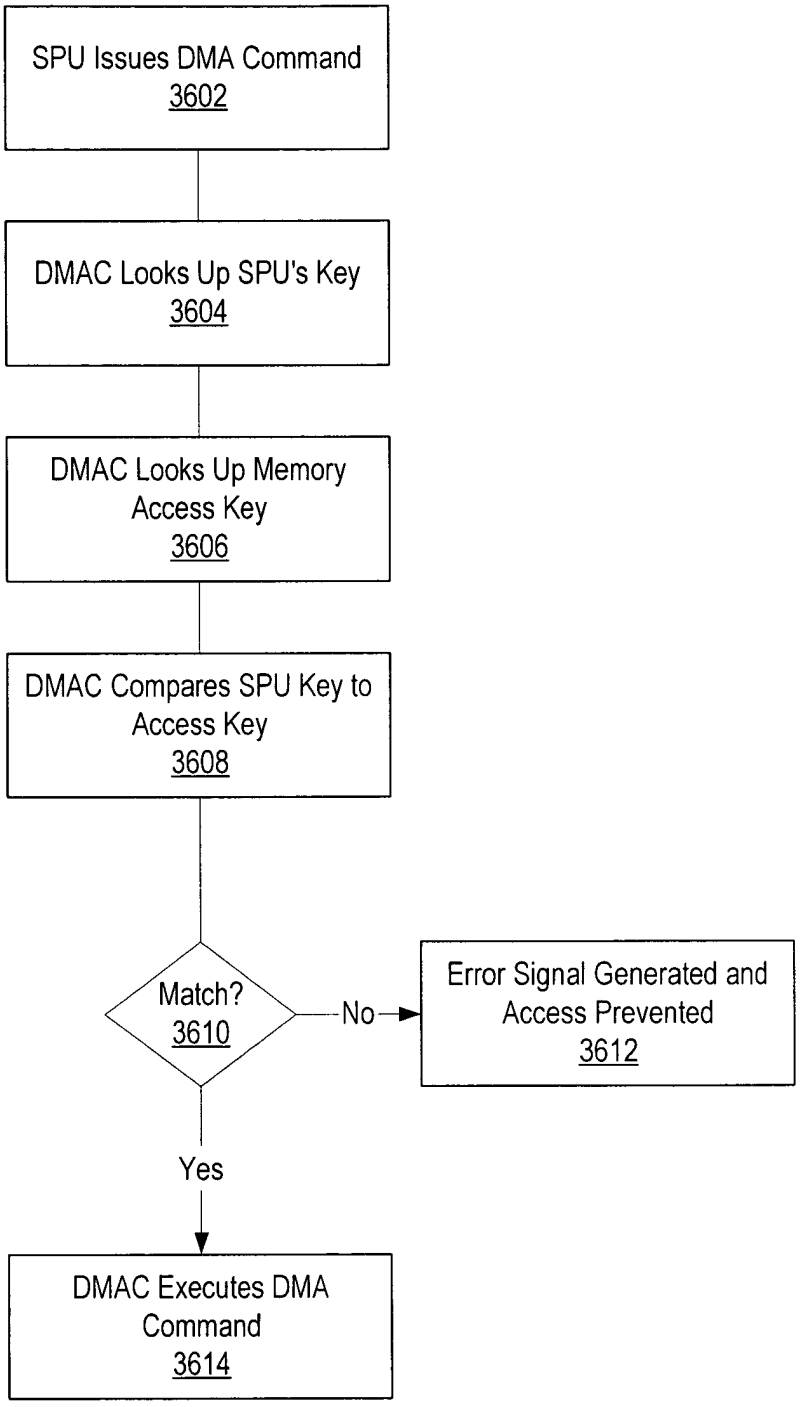
SPU Issues DMA Command
3602
DMAC Looks Up SPU's Key
3604
DMAC Looks Up Memory Access Key
3606
DMAC Compares SPU Key to Access Key
3608
Match?
3610
No
Error Signal Generated and Access Prevented
3612
Yes
DMAC Executes DMA Command
3614

Figure 36

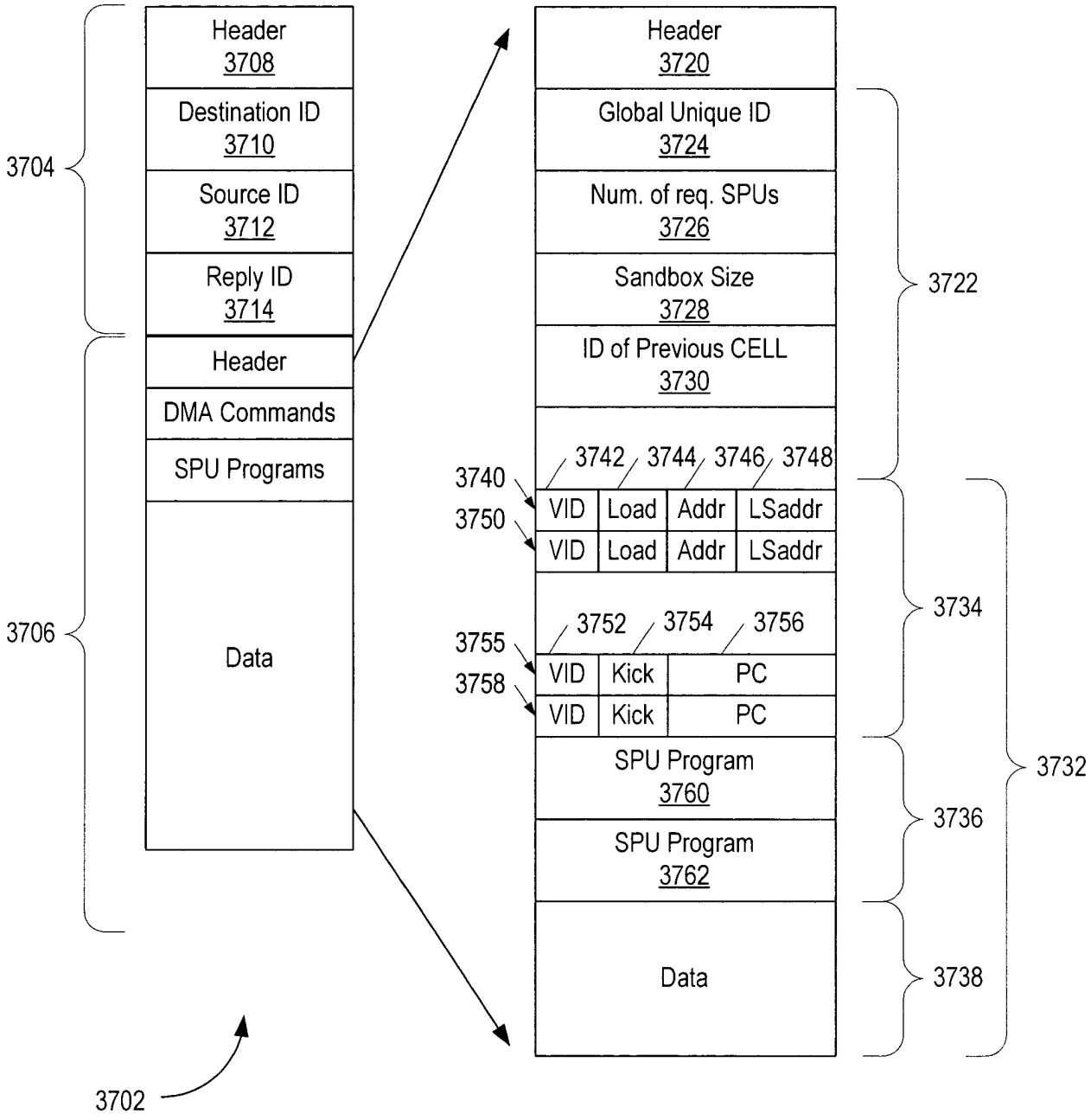
Header
3708
Destination ID
3710
Source ID
3712
Reply ID
3714
Header
DMA Commands
SPU Programs
Data
3704
3706
3702
Header
3720
Global Unique ID
3724
Num. of req. SPUs
3726
Sandbox Size
3728
ID of Previous CELL
3730
3722
3742
3744
3746
3748
3740
3750
VID
Load
Addr
LSaddr
VID
Load
Addr
LSaddr
3734
3752
3754
3756
3755
3758
VID
Kick
PC
VID
Kick
PC
SPU Program
3760
SPU Program
3762
3736
Data
3738
3732

Figure 37

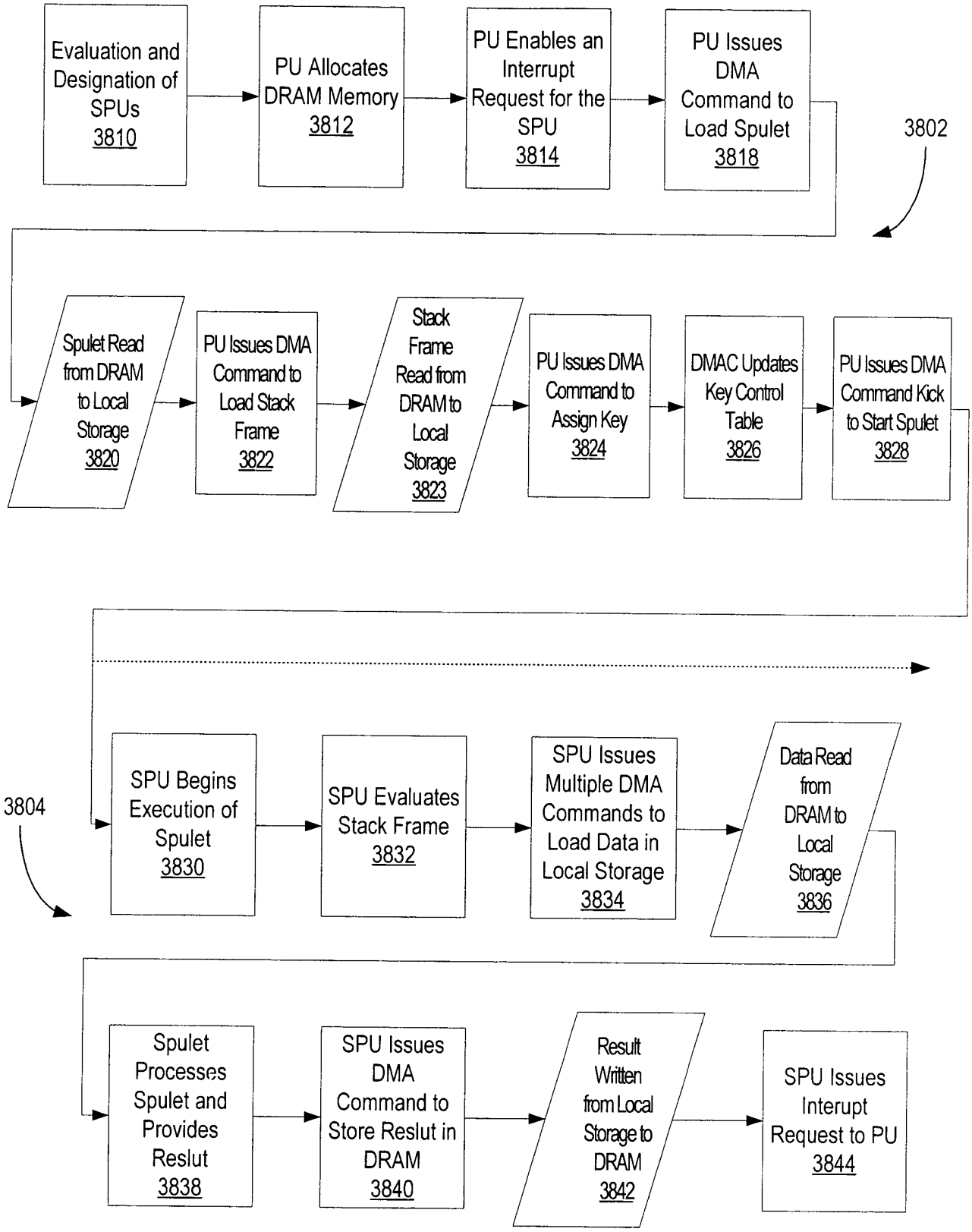
Evaluation and Designation of SPUs 3810
PU Allocates DRAM Memory 3812
PU Enables an Interrupt Request for the SPU 3814
PU Issues DMA Command to Load Spulet 3818
3802
Spulet Read from DRAM to Local Storage 3820
PU Issues DMA Command to Load Stack Frame 3822
Stack Frame Read from DRAM to Local Storage 3823
PU Issues DMA Command to Assign Key 3824
DMAC Updates Key Control Table 3826
PU Issues DMA Command Kick to Start Spulet 3828
3804
SPU Begins Execution of Spulet 3830
SPU Evaluates Stack Frame 3832
SPU Issues Multiple DMA Commands to Load Data in Local Storage 3834
Data Read from DRAM to Local Storage 3836
Spulet Processes Spulet and Provides Reslut 3838
SPU Issues DMA Command to Store Reslut in DRAM 3840
Result Written from Local Storage to DRAM 3842
SPU Issues Interupt Request to PU 3844

Figure 38

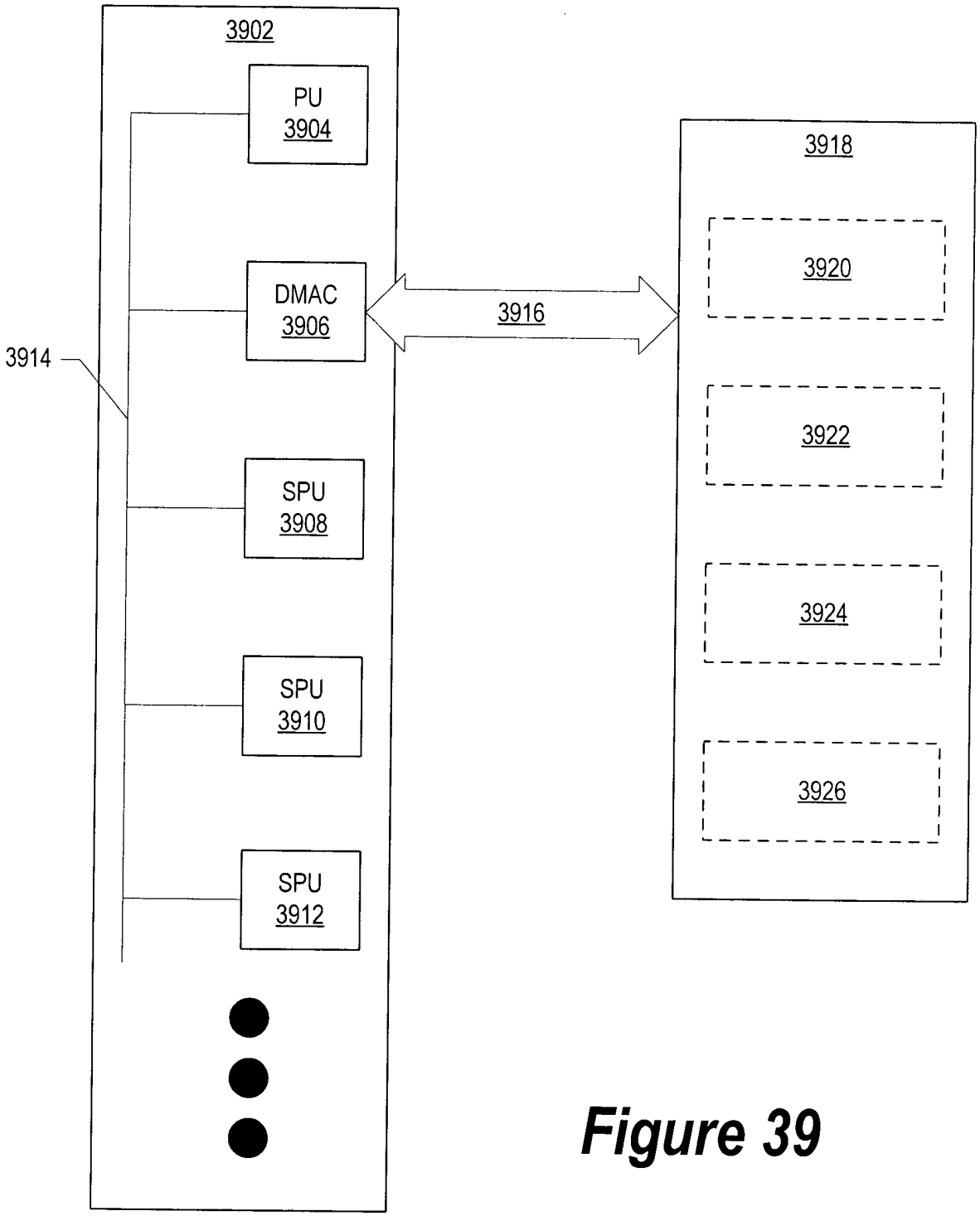
3902
PU
3904
DMAC
3906
3914
SPU
3908
SPU
3910
SPU
3912
3916
3918
3920
3922
3924
3926

Figure 39

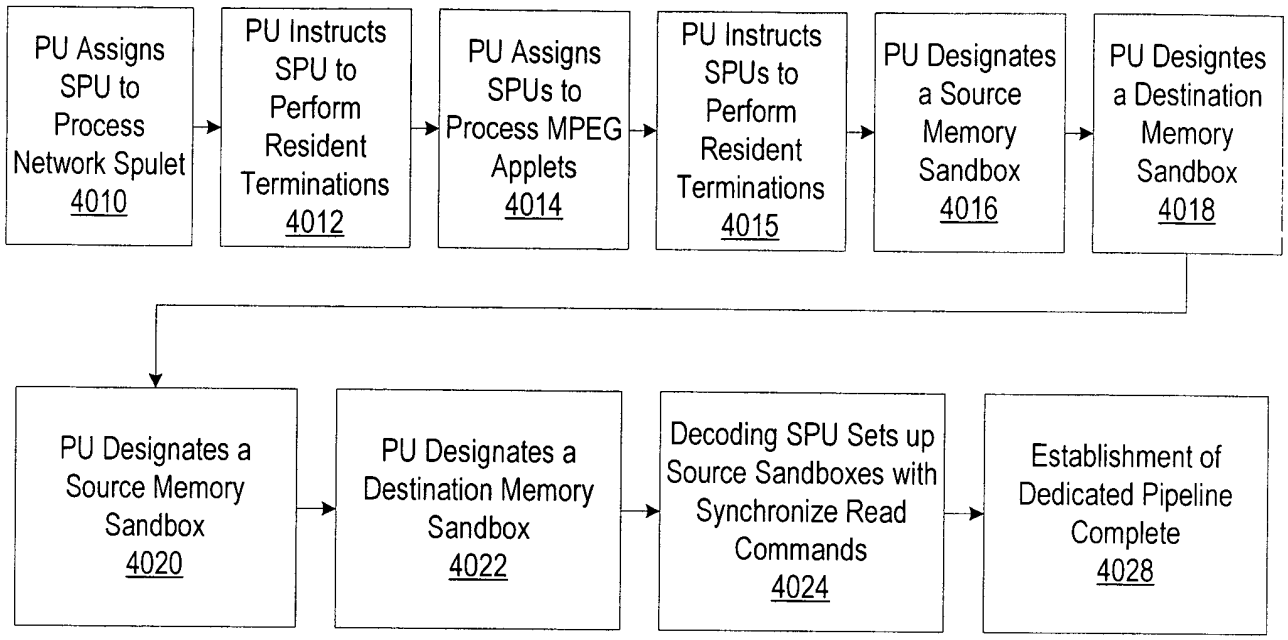

Figure 40A

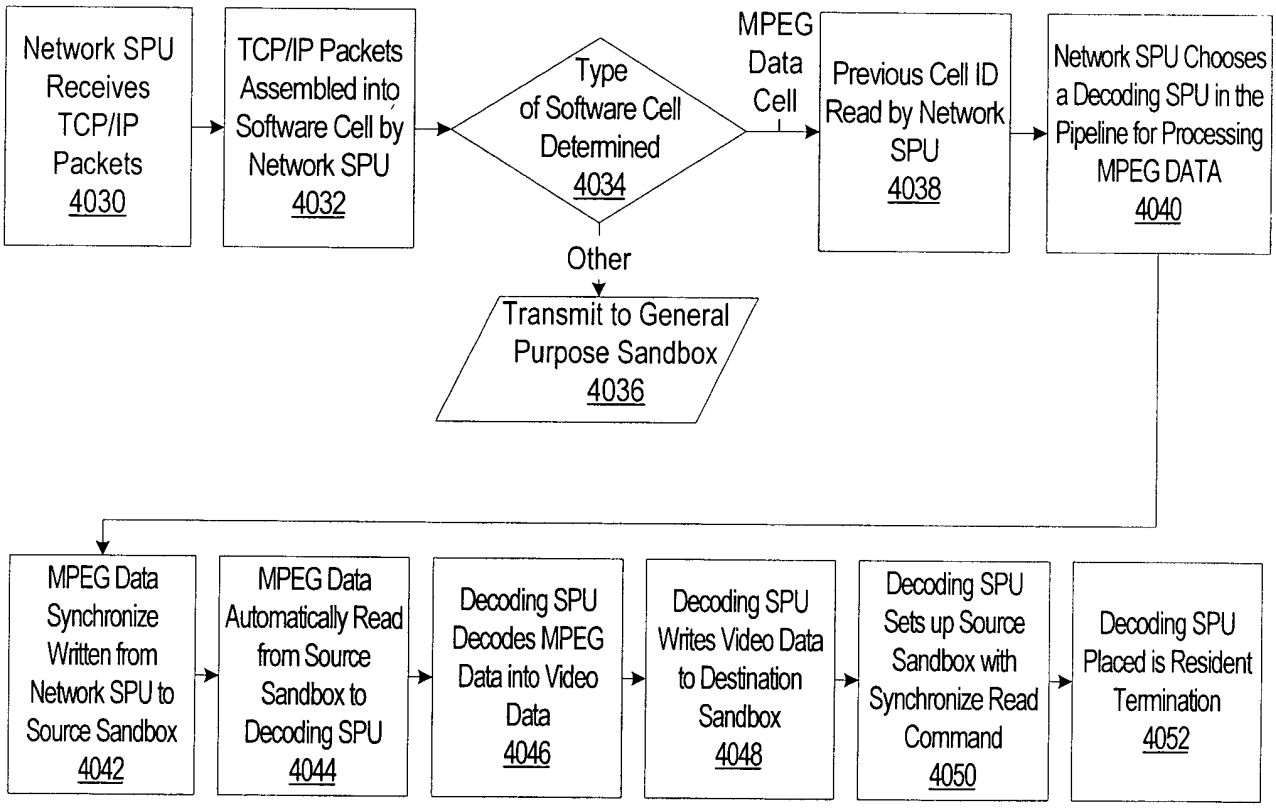

Figure 40B

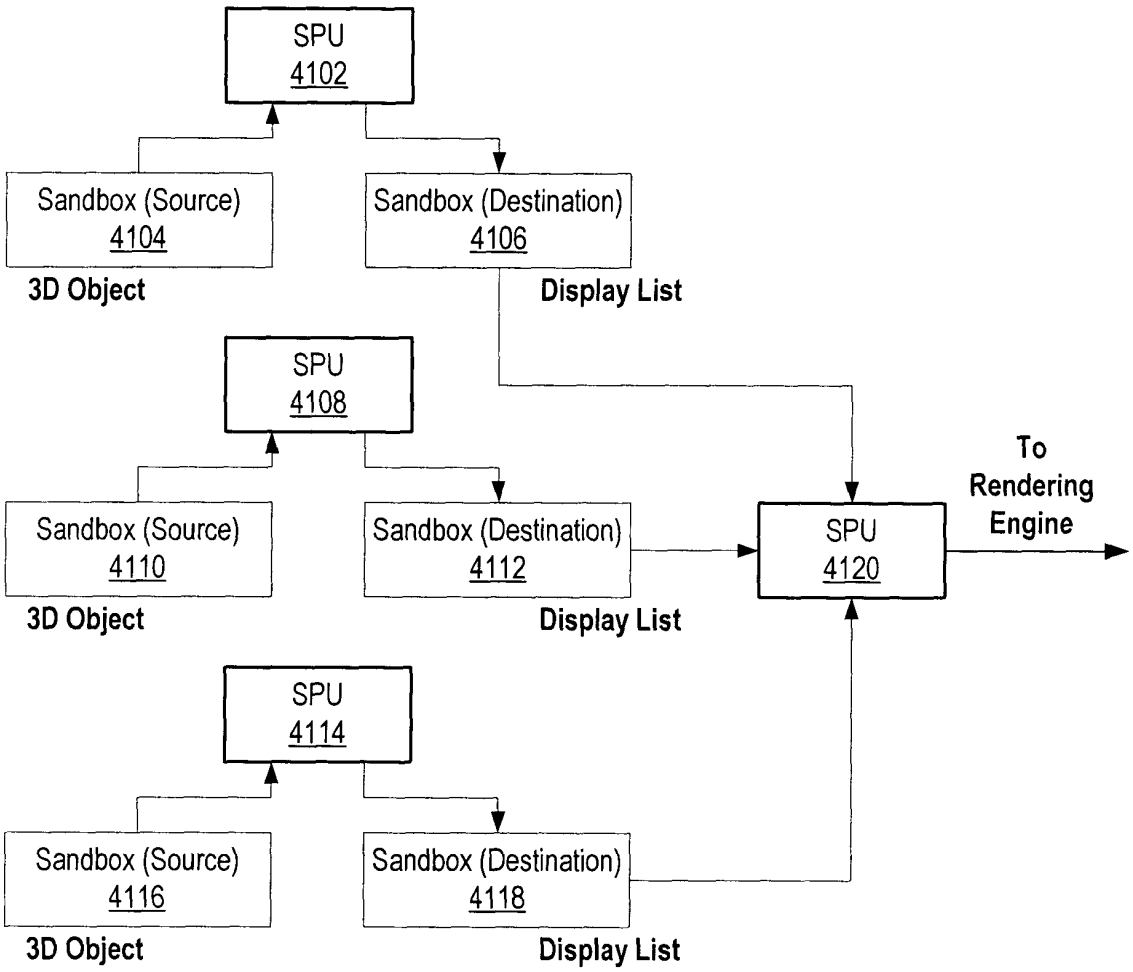
SPU
4102
Sandbox (Source)
4104
Sandbox (Destination)
4106
3D Object
Display List
SPU
4108
Sandbox (Source)
4110
Sandbox (Destination)
4112
3D Object
Display List
SPU
4120
To
Rendering
Engine
SPU
4114
Sandbox (Source)
4116
Sandbox (Destination)
4118
3D Object
Display List

Figure 41

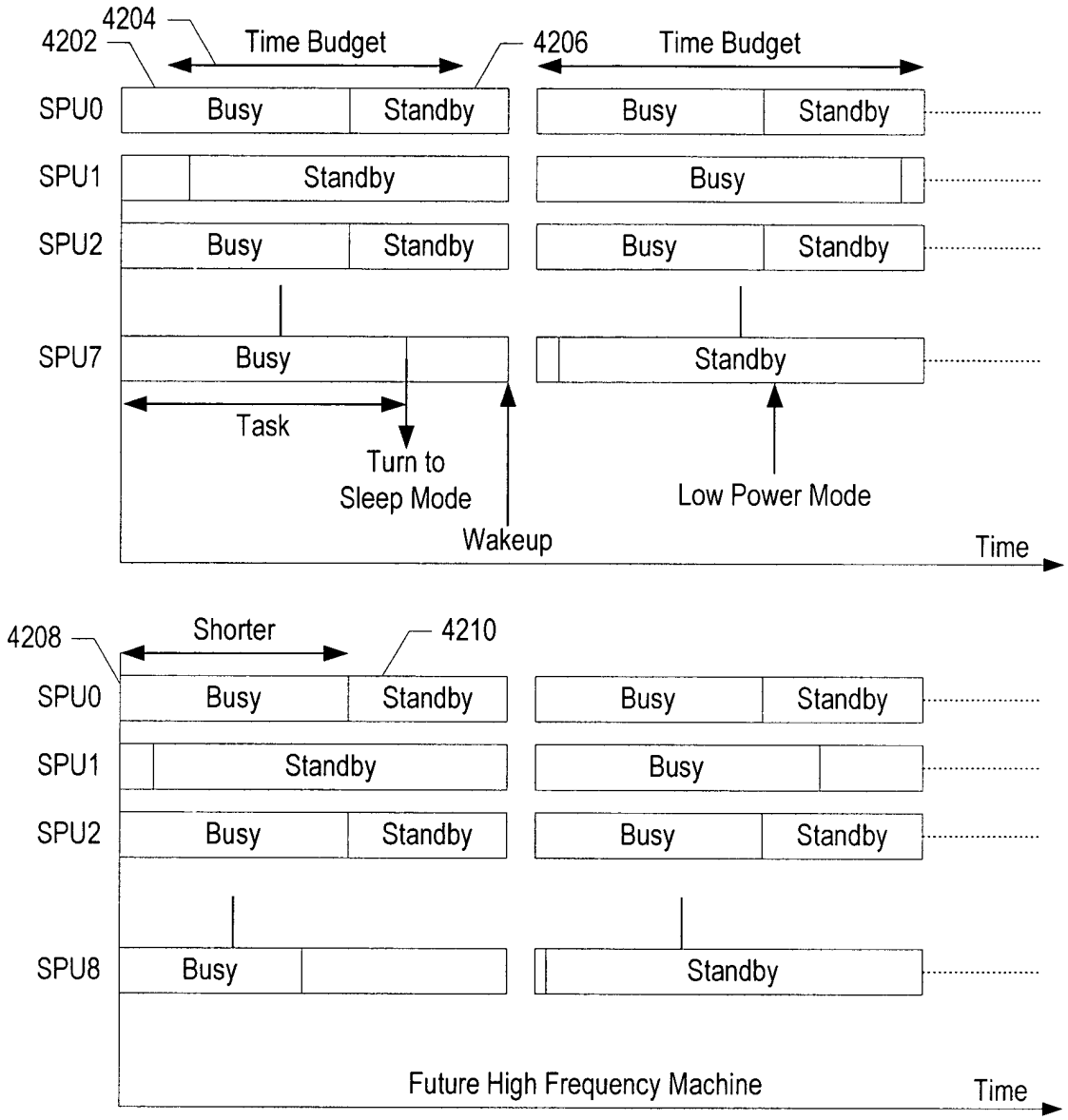
4204
4202
Time Budget
4206
Time Budget
SPU0
Busy
Standby
Busy
Standby
SPU1
Standby
Busy
SPU2
Busy
Standby
Busy
Standby
SPU7
Busy
Standby
Task
Turn to
Sleep Mode
Low Power Mode
Wakeup
Time
4208
Shorter
4210
SPU0
Busy
Standby
Busy
Standby
SPU1
Standby
Busy
SPU2
Busy
Standby
Busy
Standby
SPU8
Busy
Standby
Future High Frequency Machine
Time

Figure 42

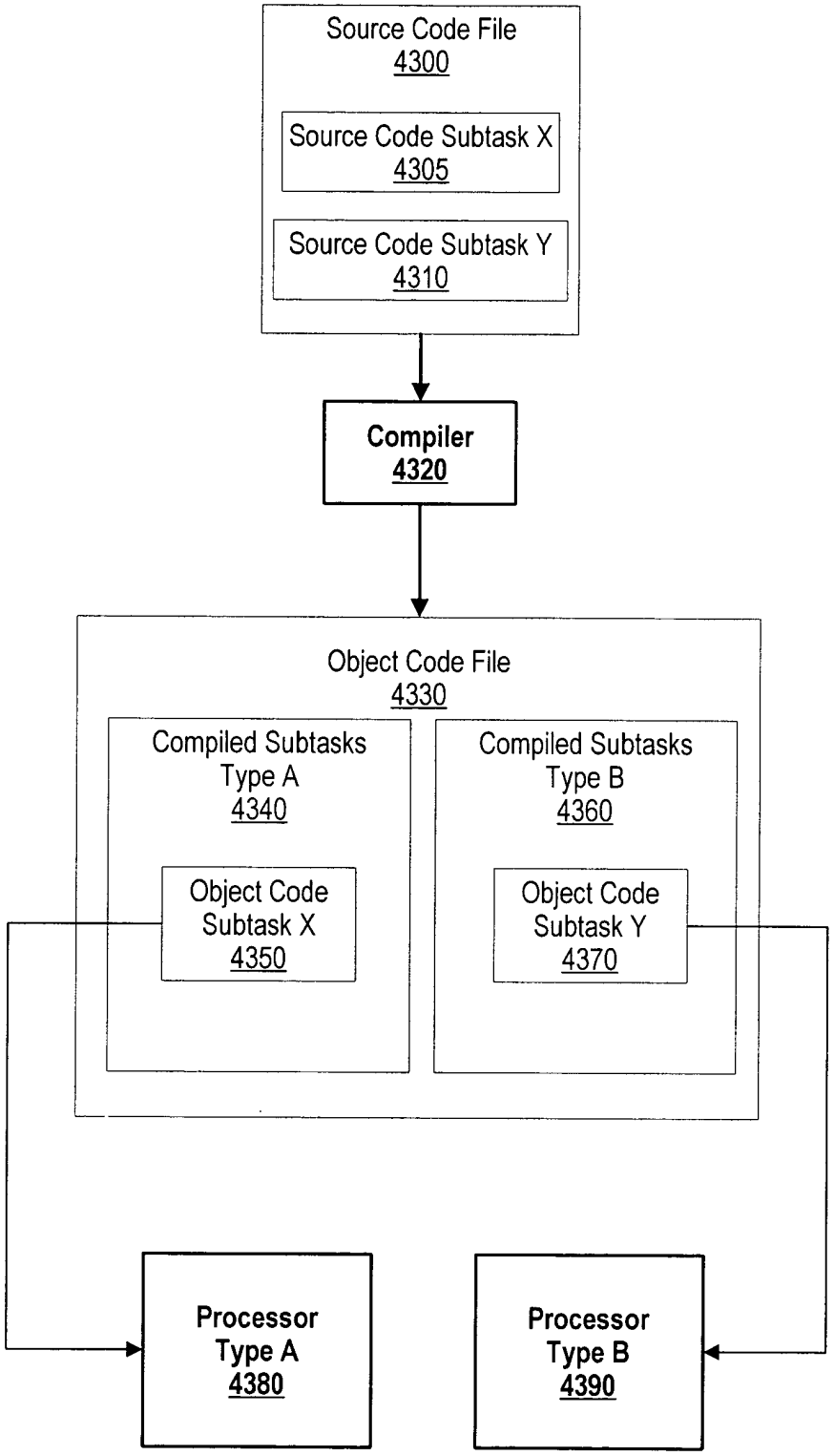
Source Code File
4300
Source Code Subtask X
4305
Source Code Subtask Y
4310
Compiler
4320
Object Code File
4330
Compiled Subtasks
Type A
4340
Object Code
Subtask X
4350
Compiled Subtasks
Type B
4360
Object Code
Subtask Y
4370
Processor
Type A
4380
Processor
Type B
4390

Figure 43

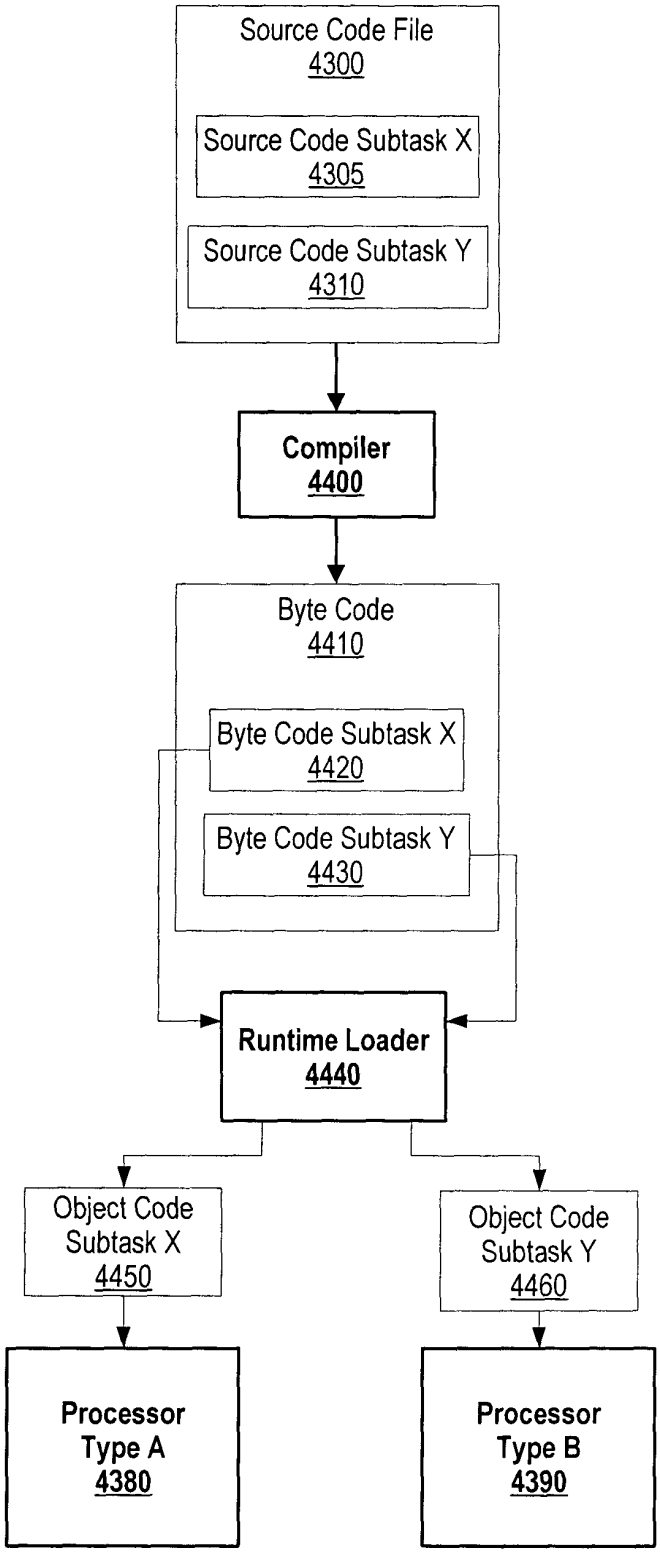
Source Code File
4300
Source Code Subtask X
4305
Source Code Subtask Y
4310
Compiler
4400
Byte Code
4410
Byte Code Subtask X
4420
Byte Code Subtask Y
4430
Runtime Loader
4440
Object Code
Subtask X
4450
Object Code
Subtask Y
4460
Processor
Type A
4380
Processor
Type B
4390

Figure 44

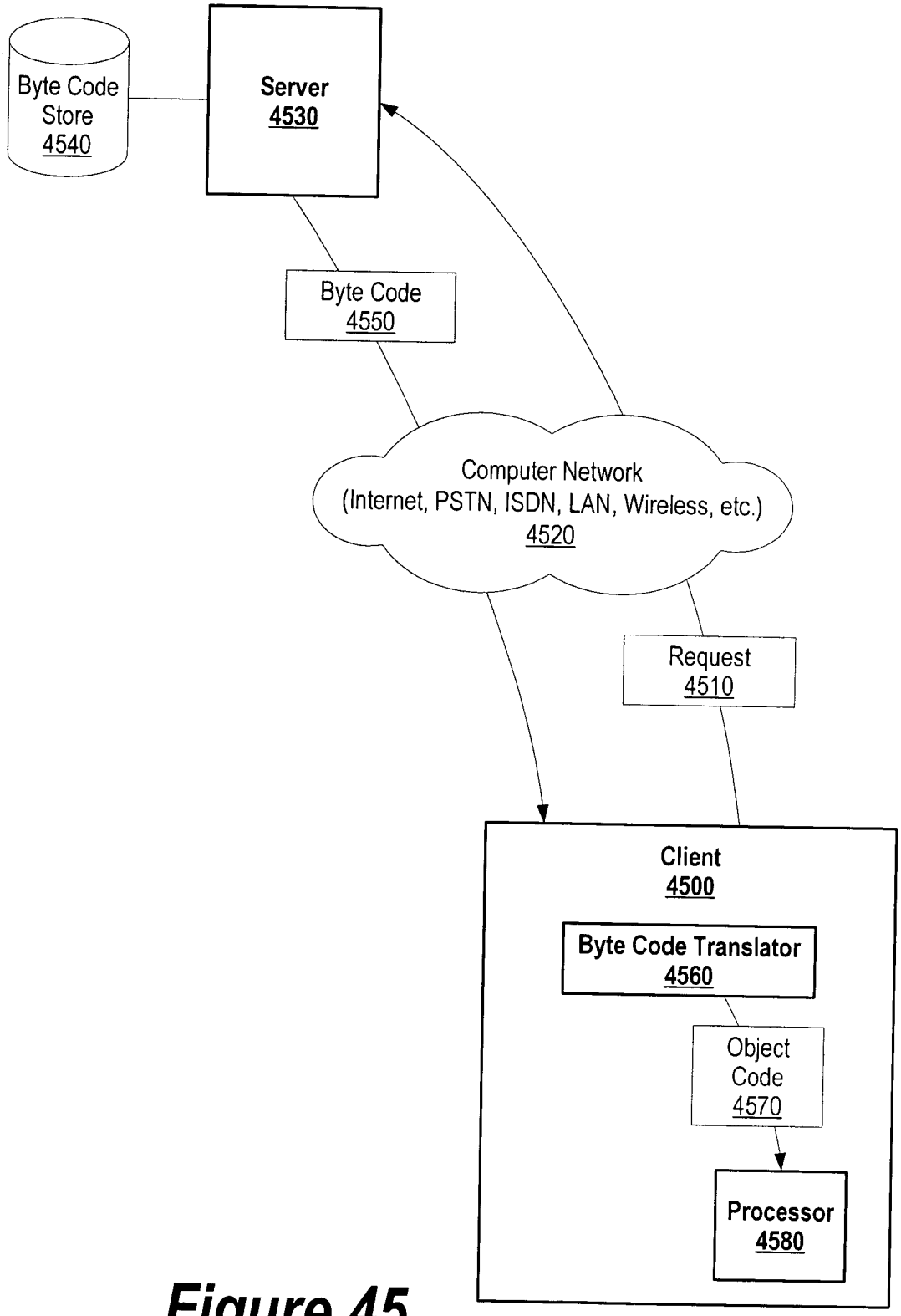
Byte Code Store 4540
Server 4530
Byte Code 4550
Computer Network (Internet, PSTN, ISDN, LAN, Wireless, etc.) 4520
Request 4510
Client 4500
Byte Code Translator 4560
Object Code 4570
Processor 4580

Figure 45

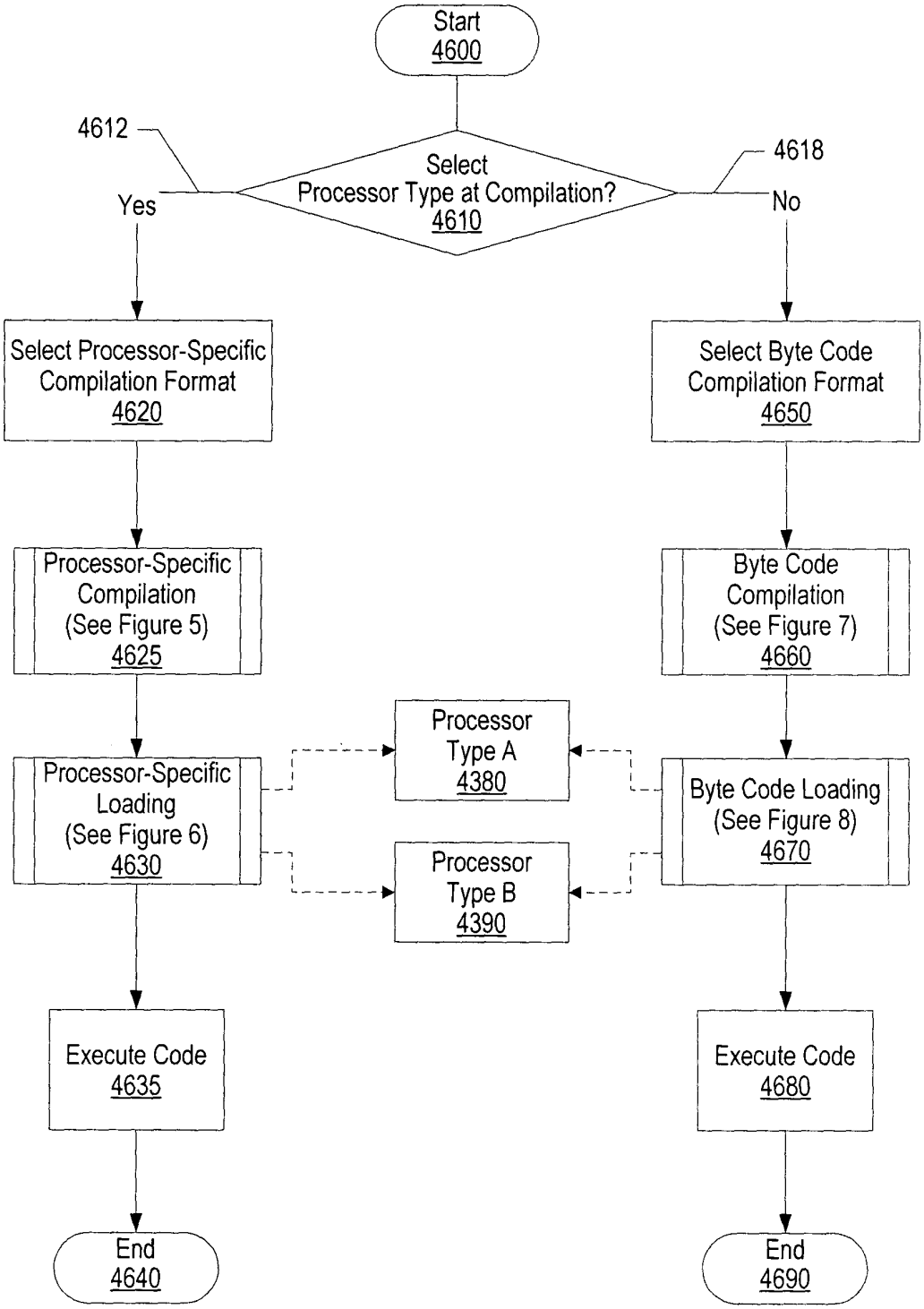
Start
4600
4612
Select
Processor Type at Compilation?
4610
4618
Yes
No
Select Processor-Specific
Compilation Format
4620
Select Byte Code
Compilation Format
4650
Processor-Specific
Compilation
(See Figure 5)
4625
Byte Code
Compilation
(See Figure 7)
4660
Processor
Type A
4380
Processor-Specific
Loading
(See Figure 6)
4630
Byte Code Loading
(See Figure 8)
4670
Processor
Type B
4390
Execute Code
4635
Execute Code
4680
End
4640
End
4690

Figure 46

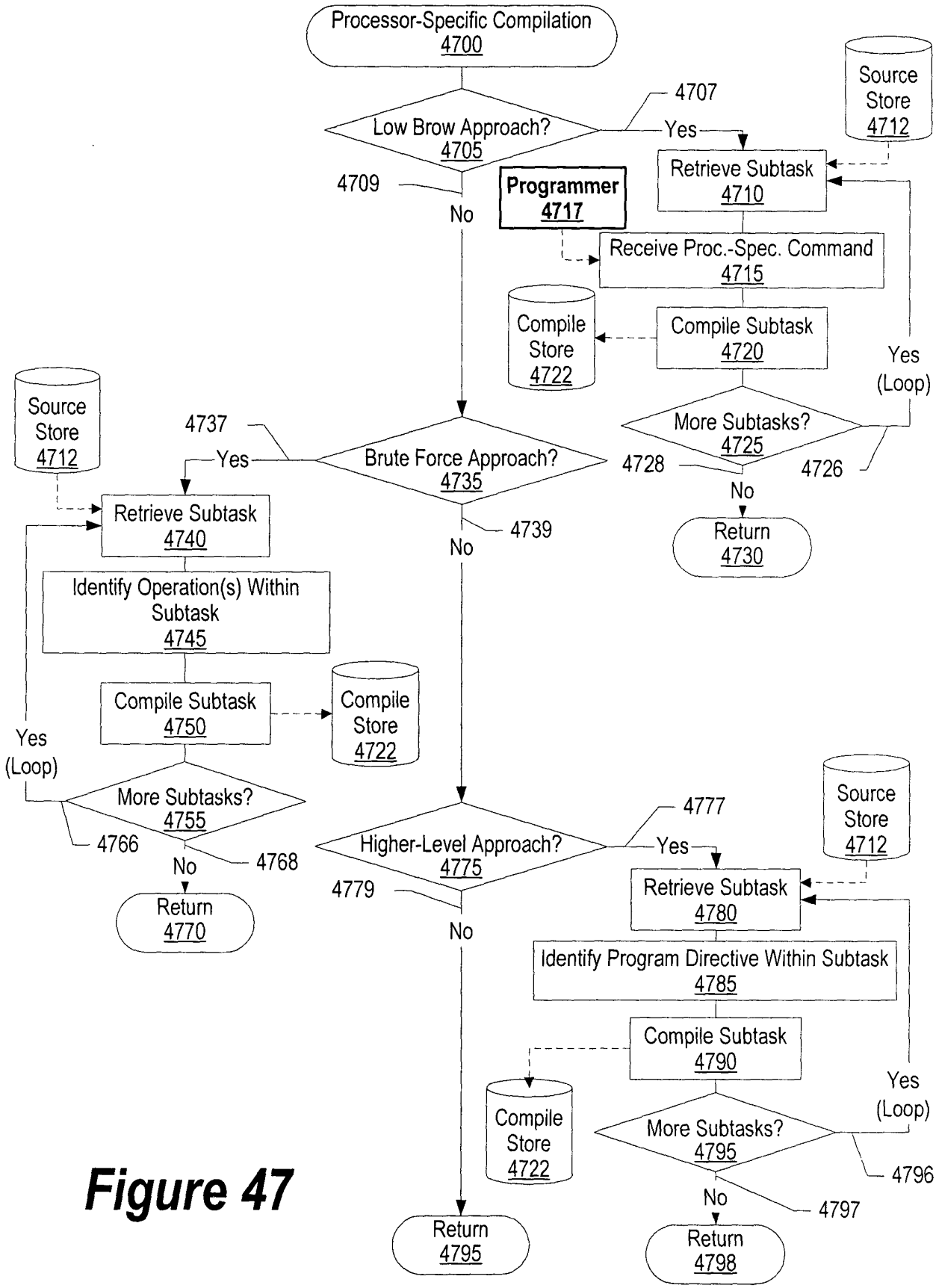
Processor-Specific Compilation
4700
Low Brow Approach?
4705
4707
Yes
4709
No
Programmer
4717
Source Store
4712
Retrieve Subtask
4710
Receive Proc.-Spec. Command
4715
Compile Store
4722
Compile Subtask
4720
Yes
(Loop)
More Subtasks?
4725
4728
4726
No
Return
4730
Source Store
4712
4737
Yes
Brute Force Approach?
4735
4739
No
Retrieve Subtask
4740
Identify Operation(s) Within Subtask
4745
Compile Subtask
4750
Compile Store
4722
Yes
(Loop)
More Subtasks?
4755
4766
4768
No
Return
4770
Higher-Level Approach?
4775
4777
Yes
4779
No
Source Store
4712
Retrieve Subtask
4780
Identify Program Directive Within Subtask
4785
Compile Subtask
4790
Compile Store
4722
Yes
(Loop)
More Subtasks?
4795
4796
No
4797
Return
4795
Return
4798

Figure 47

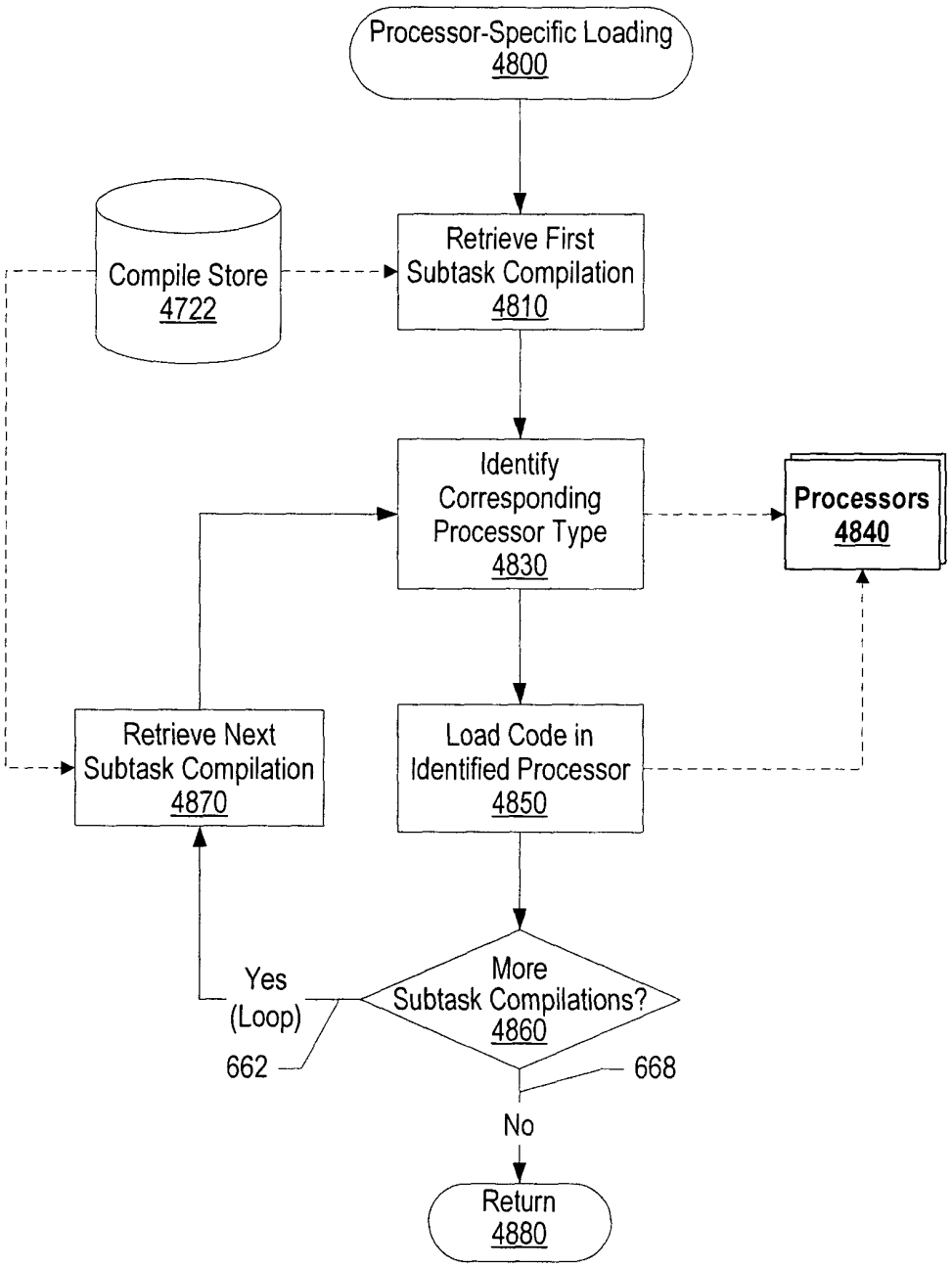
Processor-Specific Loading
4800
Compile Store
4722
Retrieve First
Subtask Compilation
4810
Identify
Corresponding
Processor Type
4830
Processors
4840
Retrieve Next
Subtask Compilation
4870
Load Code in
Identified Processor
4850
More
Subtask Compilations?
4860
Yes
(Loop)
662
668
No
Return
4880

Figure 48

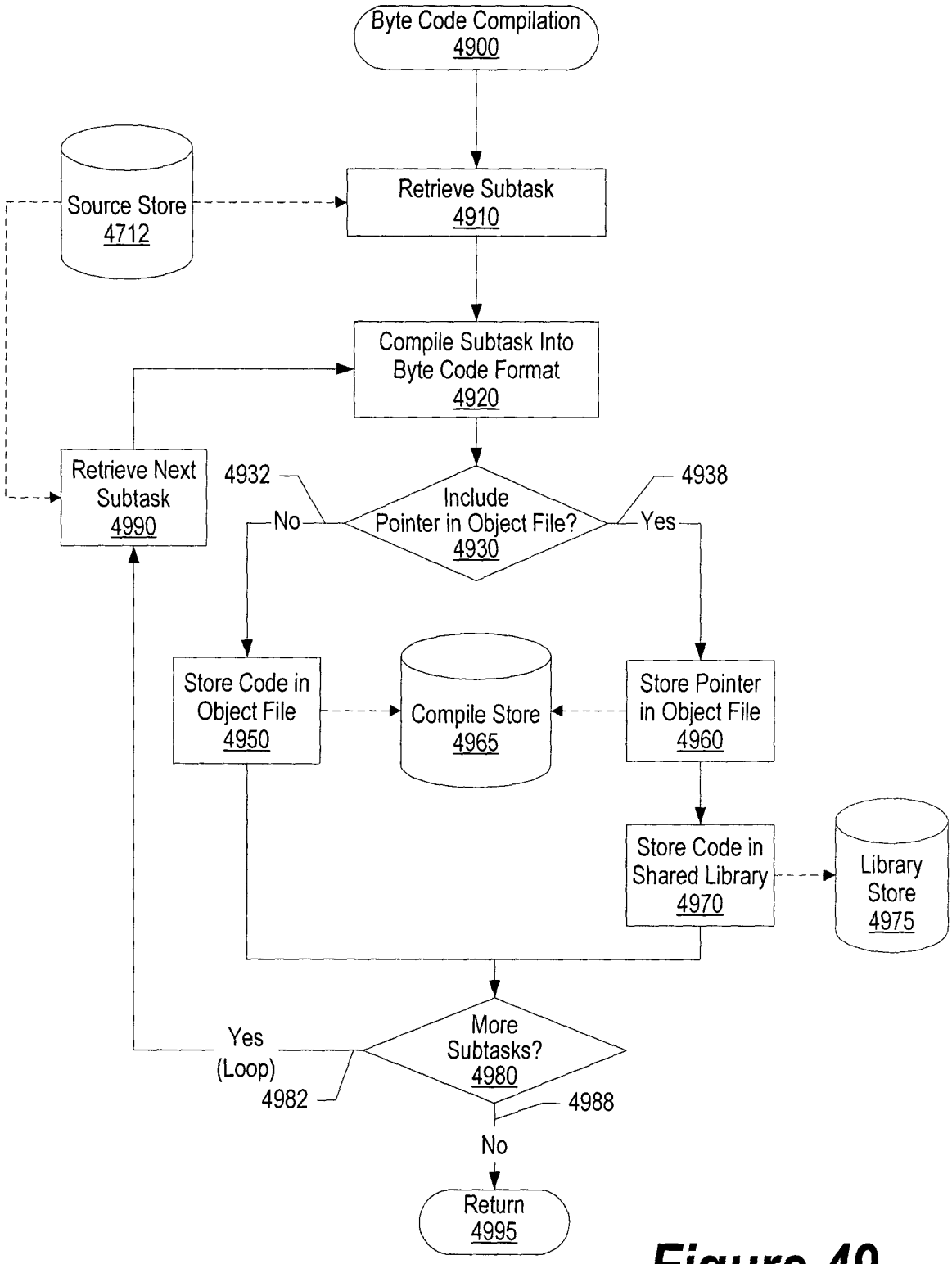
Byte Code Compilation
4900
Source Store
4712
Retrieve Subtask
4910
Compile Subtask Into
Byte Code Format
4920
Retrieve Next
Subtask
4990
4932
No
Include
Pointer in Object File?
4930
Yes
4938
Store Code in
Object File
4950
Compile Store
4965
Store Pointer
in Object File
4960
Store Code in
Shared Library
4970
Library
Store
4975
More
Subtasks?
4980
Yes
(Loop)
4982
4988
No
Return
4995

Figure 49

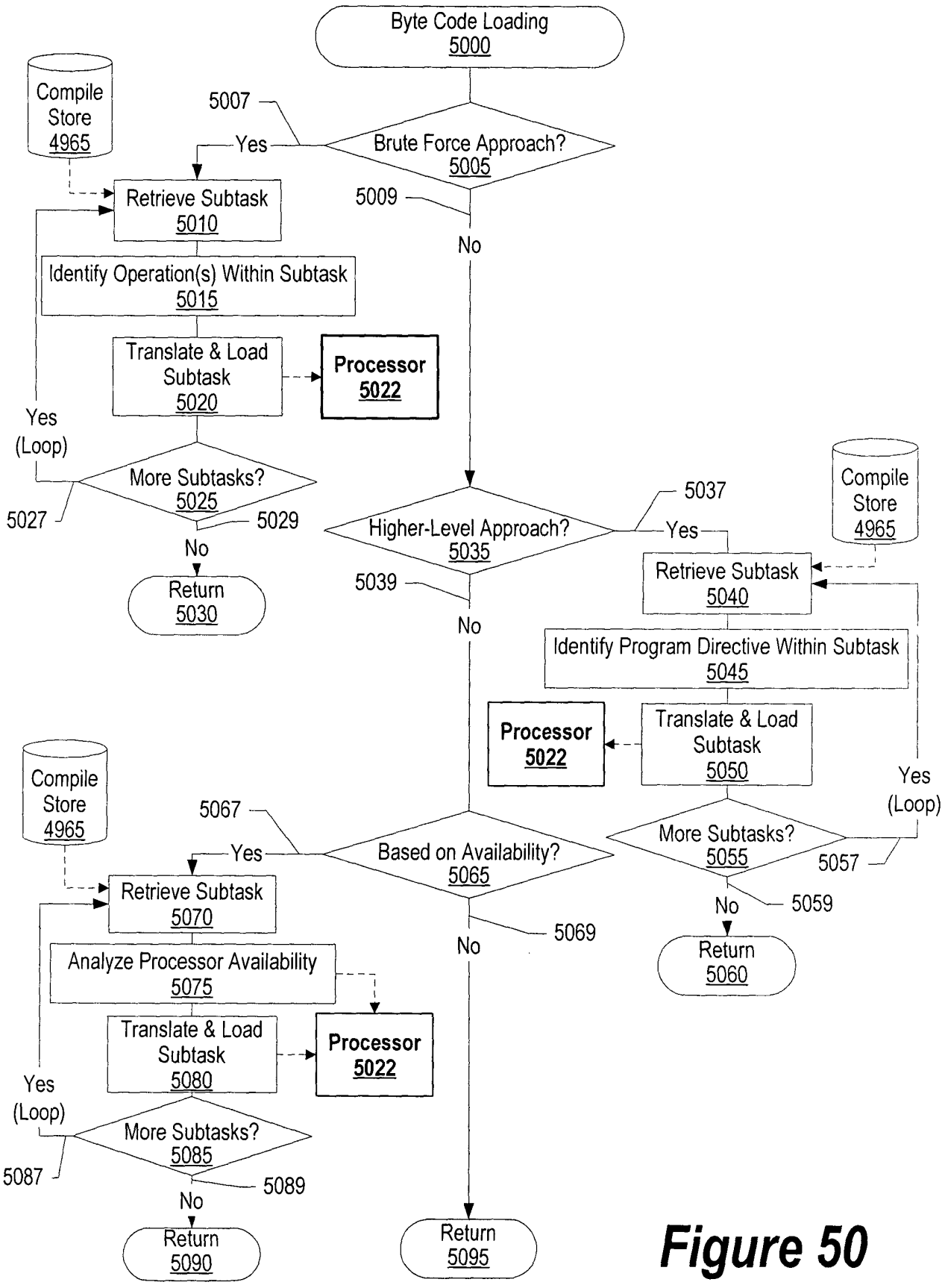
Byte Code Loading
5000
Compile Store 4965
5007
Yes
Brute Force Approach?
5005
5009
No
Retrieve Subtask
5010
Identify Operation(s) Within Subtask
5015
Translate & Load Subtask
5020
Processor
5022
Yes
(Loop)
More Subtasks?
5025
5027
5029
No
Return
5030
Higher-Level Approach?
5035
5037
Yes
5039
No
Compile Store 4965
Retrieve Subtask
5040
Identify Program Directive Within Subtask
5045
Processor
5022
Translate & Load Subtask
5050
More Subtasks?
5055
Yes
(Loop)
5057
No
5059
Return
5060
Compile Store 4965
5067
Yes
Based on Availability?
5065
5069
No
Retrieve Subtask
5070
Analyze Processor Availability
5075
Translate & Load Subtask
5080
Processor
5022
Yes
(Loop)
More Subtasks?
5085
5087
5089
No
Return
5090
Return
5095

Figure 50

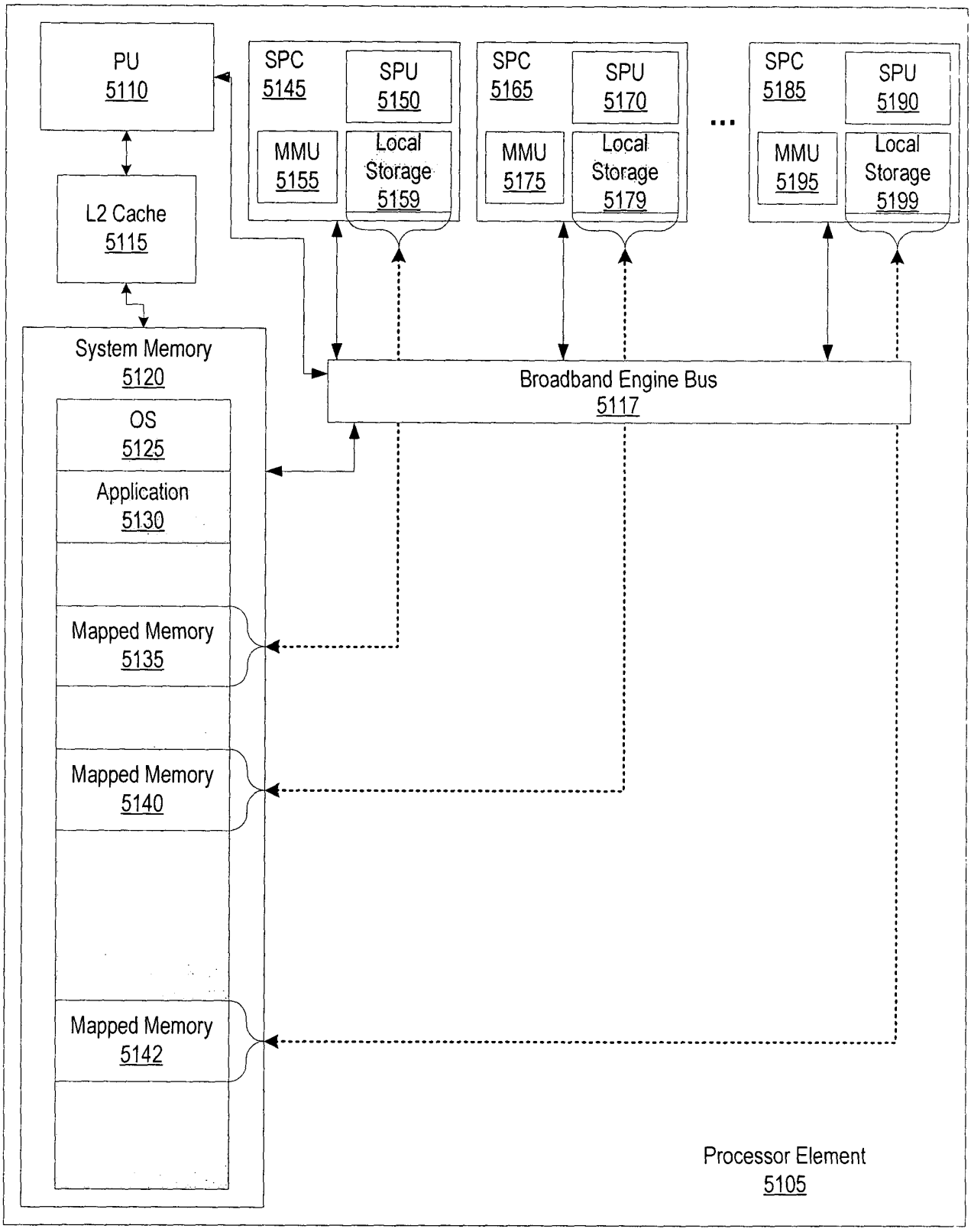
PU
5110
L2 Cache
5115
SPC
5145
SPU
5150
MMU
5155
Local Storage
5159
SPC
5165
SPU
5170
MMU
5175
Local Storage
5179
...
SPC
5185
SPU
5190
MMU
5195
Local Storage
5199
System Memory
5120
OS
5125
Application
5130
Mapped Memory
5135
Mapped Memory
5140
Mapped Memory
5142
Broadband Engine Bus
5117
Processor Element
5105

Figure 51